A SOFTWARE QUALITY MODEL FOR ANDROID APPLICATIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

MERVE VİLDAN ŞİMŞEK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE
OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

FEBRUARY 2016

A SOFTWARE QUALITY MODEL FOR ANDROID APPLICATIONS

Submitted by **MERVE VİLDAN ŞİMŞEK** in partial fulfillment of the requirements for the degree of **Master of Science in Information Systems, Middle East Technical University** by,

Prof. Dr. Nazife Baykal _____
Director, Informatics Institute

Prof. Dr. Yasemin Yardımcı Çetin _____
Head of Department, Information Systems

Doç. Dr. Aysu Betin Can _____
Supervisor, Information Systems, METU

Examining Committee Members

Prof. Dr. Nazife Baykal
IS, Middle East Technical University _____

Assoc. Prof. Dr. Aysu Betin Can
IS, Middle East Technical University _____

Assist. Prof. Dr. Erhan Eren
IS, Middle East Technical University _____

Assist. Prof. Dr. Sadık Eşmelioğlu
CENG, Çankaya University _____

Assist. Prof. Dr. Abdül Kadir Görür
CENG, Çankaya University _____

Date: 05.02.2016

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: Merve Vildan ŞİMŞEK

Signature:

ABSTRACT

A SOFTWARE QUALITY MODEL FOR ANDROID APPLICATIONS

Şimşek, Merve Vildan

M.Sc., Department of Information Systems

Advisor: Assoc. Prof. Dr. Aysu Betin Can

FEBRUARY 2016, 149 Pages

Nowadays mobile devices have become increasingly widespread. It causes mobile applications number to grow dramatically. As the popularity of these systems is predicted to continue its increase in the near future, the importance of the quality of mobile applications increases. The aim of this study is to present a quality model for Android applications. We chose applications developed for Android Operating System as our target because of its prevalence in the mobile market. To achieve the aim of the study, we analyzed traditional software quality characteristics, which are described in ISO/IEC SQuaRE Software Quality Standard and selected applicable quality characteristics. Afterwards, we have identified new Android-specific source code metrics and quality characteristics. We have developed a quality model that contains the resulting quality characteristics and applied this model to Android applications in a case study to show the applicability of the model.

Keywords: Software Quality, Mobile Software, Quality Model, Android Applications, ISO/IEC SQuaRE Standard

ÖZ

ANDROID UYGULAMALAR İÇİN YAZILIM KALİTE MODELİ

Şimşek, Merve Vildan

Yüksek Lisans, Bilişim Sistemleri Bölümü

Tez Danışmanı: Doç. Dr. Aysu Betin Can

Şubat 2016, 149 Sayfa

Günümüzde akıllı telefon ve tablet gibi mobil cihazlar giderek yaygınlaşmaktadır. Bu durum mobil uygulama sayısının önemli ölçüde artışını da beraberinde getirmektedir. Bu sistemlerin popülerliğini yakın gelecekte de arttırmaya devam edeceği tahmin edildiğinden, mobil uygulamaların kalitesi oldukça fazla önem kazanmaktadır. Bu çalışmanın amacı Android uygulamaları için bir kalite modeli oluşturulmasıdır. Mobil pazardaki yaygınlıklarından dolayı Android İşletim Sistemi için geliştirilmiş uygulamalar hedef olarak seçildi. Çalışmanın amacına ulaşması için ISO/IEC SQuaRE Yazılım Kalite Standardında tanımlanmış olan geleneksel yazılım kalite özellikleri analiz edildi ve uygulanabilir kalite özellikleri belirlendi. Sonrasında Android uygulamalara özgü yeni kaynak kod metrikleri ve kalite özelliklerini belirlendi. Nihai özelliklerin yer aldığı bir kalite modeli oluşturuldu ve bu model bir örnek olay incelemesinde Android uygulamalarına uygulanarak, modelin uygulanabilirliği gösterildi.

Anahtar Kelimeler: Yazılım Kalitesi, Mobil Yazılım, Kalite Modeli, Android Uygulamalar, ISO/IEC SQuaRE Standardı

To My Family
Tacettin, Oya Selma, Ahsen and Muhsin

ACKNOWLEDGEMENTS

I would like to express my gratitude to my advisor Assoc. Prof. Dr. AYSU BETİN CAN for her guidance, encouragements and support during my thesis study.

I also would like to thank my amazing family: Tacettin, Oya Selma, Ahsen and Safa Muhsin ŞİMŞEK and my amazing grandfather: Muhsin ŞİMŞEK for their concern during this process.

I would like to send a special thanks to my best friend Gülçin HÖKELEKLİ for all of her help and support throughout Master's Program.

TABLE OF CONTENTS

ABSTRACT vi

ÖZ vii

ACKNOWLEDGEMENTS ix

TABLE OF CONTENTS xi

LIST OF TABLES xiii

LIST OF FIGURES xiv

LIST OF ABBREVIATIONS xv

1 INTRODUCTION 1

2 LITERATURE REVIEW 3

2.1 SOFTWARE QUALITY MODELS 3

2.1.1 McCall's Model (1977) 3

2.1.2 Boehm's Quality Model (1978) 4

2.1.3 FURPS (1987) 5

2.1.4 ISO/IEC 9126 Standard Quality Model (1991) 6

2.1.5 Dromey's Quality Model (1995) 7

2.1.6 Bansiya's QMOOD Model (2002) 8

2.1.7 ISO/IEC 25000 Standard (SQuaRE's Model) (2011) 9

2.2 QUALITY MODELS FOR MOBILE APPLICATIONS 10

3 ANDROID QUALITY MODEL DEVELOPMENT 13

3.1 IDENTIFYING QUALITY CHARACTERISTICS 14

3.2 IDENTIFYING METRICS 17

3.3 METRICS - QUALITY CHARACTERISTICS RELATIONSHIP 20

3.3.1 Weighting The Relationship 21

4 MODEL EVALUATION & RESULTS 25

4.1 MODEL VALIDATION 25

4.1.1 Determination of a Rating Scale 29

4.3 GATHERING EXPERIMENT DATA FOR CASE STUDY 30

4.4 GATHERING AND NORMALIZING METRIC DATA 31

4.5 ANALYZING RESULTS 36

5 CONCLUSION AND FUTURE WORK 55

5.1 CONCLUSION 55

5.2 LIMITATIONS 55

5.3 FUTURE WORK 56

REFERENCES 57

APPENDICES 61

Appendix A: Survey Questions 61

Appendix B: Ethics Approval Form **72**
Appendix C: Summary of Survey Results **74**
Appendix D: Cronbach's Alpha Values of Survey Results **145**

LIST OF TABLES

Table 3. 1: METRICS USED FOR QUALITY ASSESSMENT FOR OBJECT-ORIENTED DESIGN IN LITERATURE 17
Table 3. 2: LIST OF METRICS USED IN OUR MODEL 19
Table 3. 3: METRICS - QUALITY CHARACTERISTICS RELATIONSHIP (THE MEDIAN OF RESPONSES) 22
Table 3. 4: COMPUTATION FORMULAS FOR QUALITY CHARACTERISTICS 23
Table 3. 5: MIN AND MAX VALUES OF QUALITY CHARACTERISTICS 24
Table 4. 1: SCORES OF THE EVALUATORS TO 2048-ANDROID APPLICATIONS 26
Table 4. 2: METRIC VALUES FOR 2048-ANDROID 26
Table 4. 3: COMPUTED QUALITY CHARACTERISTICS FOR 2048-ANDROID 27
Table 4. 4: COMPUTED QUALITY CHARACTERISTICS FOR 2048-ANDROID WITHIN THE NEW RANGE [0, 10] 27
Table 4. 5: THE CATEGORIES OF QUALITY CHARACTERISTICS OF 2048-ANDROID APPLICATIONS 30
Table 4. 6: METRIC VALUES FOR ADBLOCK PLUS 31
Table 4. 7: METRIC VALUES FOR KEEPASSDROID 32
Table 4. 8: METRIC VALUES FOR COSYDVR 33
Table 4. 9: NORMALIZED METRIC VALUES FOR ADBLOCK PLUS 34
Table 4. 10: NORMALIZED METRIC VALUES FOR KEEPASSDROID 35
Table 4. 11: NORMALIZED METRIC VALUES FOR COSYDVR 36
Table 4. 12: COMPUTED QUALITY CHARACTERISTICS FOR ADBLOCK PLUS 37
Table 4. 13: COMPUTED QUALITY CHARACTERISTICS FOR KEEPASSDROID 37
Table 4. 14: COMPUTED QUALITY CHARACTERISTICS FOR COSYDVR 38
Table 4. 15: COMPUTED QUALITY CHARACTERISTICS FOR ADBLOCK PLUS WITHIN THE RANGE [0, 10] AND THE CATEGORIES OF THE VALUES 39
Table 4. 16: COMPUTED QUALITY CHARACTERISTICS FOR KEEPASSDROID WITHIN THE RANGE [0, 10] AND THE CATEGORIES OF THE VALUES 40
Table 4. 17: COMPUTED QUALITY CHARACTERISTICS FOR COSYDVR WITHIN THE RANGE [0, 10] AND THE CATEGORIES OF THE VALUES 41

LIST OF FIGURES

Figure 2. 1: Hierarchy of Major Perspective, Quality Characteristics and Quality Criteria of McCall Quality Model [9] 4
Figure 2. 2: Hierarchy of High Level, Intermediate Level and Primitive Characteristics of Boehm's Quality Model [11] 5
Figure 2. 3: McCall, Boehm, FURPS Models 6
Figure 2. 4: Quality Model of ISO/IEC 9126 [15] 7
Figure 2. 5: Principles of Dromey's Quality Model [16] 8
Figure 2. 6: Levels in QMOOD [18] 8
Figure 2. 7: Quality Model of ISO/IEC 25010 [18] 10
Figure 2. 8: Franke et al. Mobile Applications Quality Model [19] 11
Figure 2. 9: Zahra et al. Mobile Applications Quality Model [22] 12
Figure 3. 1: The Research Methodology 13
Figure 3. 2: The Android Applications Quality Model 17
Figure 3. 3: Demographic Data of Respondents 21
Figure 4. 1: Paired Samples T-Test of the Data Sets (Scores of Evaluators & Results of the Model) 28
Figure 4. 2: Paired Samples T-Test of the Data Sets (Scores of Evaluators & Results of the Model) 29
Figure 4. 3: Plots Of Quality Characteristics Of Adblock Plus 41
Figure 4. 4: Plots Of Quality Characteristics Of KeePassDroid 46
Figure 4. 5: Plots of Quality Characteristics of CosyDVR 50
Figure 4. 6: Quality Scores of the Last Release of Adblock Plus 53
Figure 4. 7: Quality Scores of the Last Release of KeePassDroid 53
Figure 4. 8: Quality Scores of the Last Release of CosyDVR 54

LIST OF ABBREVIATIONS

IEEE	Institute of Electrical and Electronics Engineers
IT	Information Technology
ISO	International Organization for Standardization
IEC	The International Electrotechnical Commission
SQuaRE	Software product Quality Requirements and Evaluation
QMOOD	Quality Model for Object Oriented Design
API	Application Program Interface
App	Application

CHAPTER I

1 INTRODUCTION

Mobile communication occupies an important place in our life that we feel lost without a smartphone. This new concept of phobia is called Nomophobia (no-mobile-phone phobia) in psychology [1]. What is a smartphone that causes such a phobia? A smartphone is a multifunctional device that we use for not only communication but also for entertainment, business transactions, and much more. These functions are made possible by the development of mobile applications.

The increase in smartphones and tablets causes mobile application number to grow dramatically. "More than 268 billion mobile application downloads will have taken place by 2017, generating more than $77 billion in revenue and making apps one of the most popular computing tools for users across the globe" according to Gartner Group [2]. This study shows that mobile applications are becoming an essential requirement of our lives. As the number of mobile applications is increasing at a very high rate, quality of the applications is becoming an important issue.

IEEE Definition of Software Quality is "the degree to which a system, component, or process meets specified requirements" [3]. Software Quality is essential for software engineering because the aim of software engineering is to produce software products with high quality.

Quality of a software product is significant for both user and the developer. The user wants to work with good qualitative software and developer wants his/her product to be used admiringly. The quality of a mobile software product is considerably important. Developers should not underestimate the importance of mobile app quality, because of the reasons listed below:

- Quality is not only an IT problem anymore. It affects the reputation of Developer/Company directly. Users write reviews, give stars to applications. Overview of applications' ratings and reviews are accessible for every user.
- There is a very high possibility of customer churn. The number of competitors is very high in application stores. It is very easy to abandon an application and find an alternative to it.

Quality is the combination of many characteristics. These characteristics are referred as quality characteristics. The quality characteristics and their relationships are represented in Quality Models. The models are useful because they display the factors important for quality. Developers need quality models as a guideline to maintain and improve the quality of their mobile applications.

The aim of this study is to develop a quality model for Android applications. We chose applications developed for Android Operating System as our target because of its prevalence in the mobile market. According to the statistics, the number of applications available for download in Google Play Store "as of July 2015 is 1.6 million. Apple's App Store is the second largest application store with 1.5 million available applications" [4].

In the process of Android Applications Quality Model development, we have analyzed quality characteristics, which are described in ISO/IEC SQuaRE Software Quality Standard. In the Quality Standard, quality properties are categorized into eight characteristics: functional suitability, performance efficiency, reliability, security, usability, compatibility, portability and maintainability. As our quality model deals with the quality characteristics, which are relevant to the area of mobile software, we have selected the applicable quality characteristics from the quality standard and added a new characteristic named Data Integrity.

We have determined the relationship of source code metrics and quality characteristics by conducting a survey on Android Developers. We have developed our Android Applications Quality Model, validated the quality model and applied this model to different releases of three open-source Android applications in a case study to show the applicability of the model.

Few studies have been done on the development of mobile software quality models. Most of the related studies are incapable of assessing the quality of applications because they are a lack of metric determination as explained in detail in Chapter 2. On the other hand, in our study we have determined Android-specific source code metrics to make measurable the quality of mobile applications in accordance with a quotation from Galileo Galilei "Measure what is measurable, and make measurable what is not so" [5].

The contribution of this thesis study is the validated Android Applications Quality Model based on the data gathered from Android developers, and the metrics identified for the quality model.

This thesis is structured as follows:

- In Chapter 2, we presented the literature review related to software quality models.
- In Chapter 3, we explained the proposed Android Applications Quality Model's development processes in a detailed manner.
- In Chapter 4, we presented the evaluation of the proposed model and the results of the case study.
- Finally, in Chapter 5, we presented the conclusion of the thesis work and possible future work related to the proposed model.

CHAPTER II

2 LITERATURE REVIEW

Over the last 30 years, a number of quality models have been developed. Quality models represent a quantitative structure of quality. [6] In this chapter, we presented the related literature. The first section contains the detailed expression of software quality models. The second section contains information about quality models developed for mobile applications.

2.1 SOFTWARE QUALITY MODELS

Beginning with hierarchical models proposed by McCall and Boehm, a variety of software quality models have been developed. Some of which have been standardized. In this section, we summarized the most notable software quality models.

2.1.1 McCall's Model (1977)

Jim McCall presented a quality model, which is one of the most famous predecessors of today's quality models. The model is also named as the General Electrics Model. McCall developed the model for the US military in 1977. The McCall quality model has three factors, which are Product Revision, Product Operation, and Product Transition to identify the quality of a software product. [7]

Product Revision: It is related to error correction and system adaptation.
- Maintainability: the ease of finding and fixing a bug.
- Flexibility: the ease of making changes required by modification in the operating environment.
- Testability: the ease of validating the software requirements.

Product Operation: It is related to operation characteristics.
- Correctness: the functionality conforms the specification.
- Efficiency: usage of system resources such as storage, network, processor time.
- Integrity: protecting a system from unauthorized access.
- Reliability: the system's ability not to fail.
- Usability: the ease of use of the software.

Product Transition: It is related adaptability to rapid changes in hardware.
- Reusability: the ability to use software components in a different context.
- Portability: the ease of transferring the software from one environment to another.

- Interoperability: the ability of software components to work together.

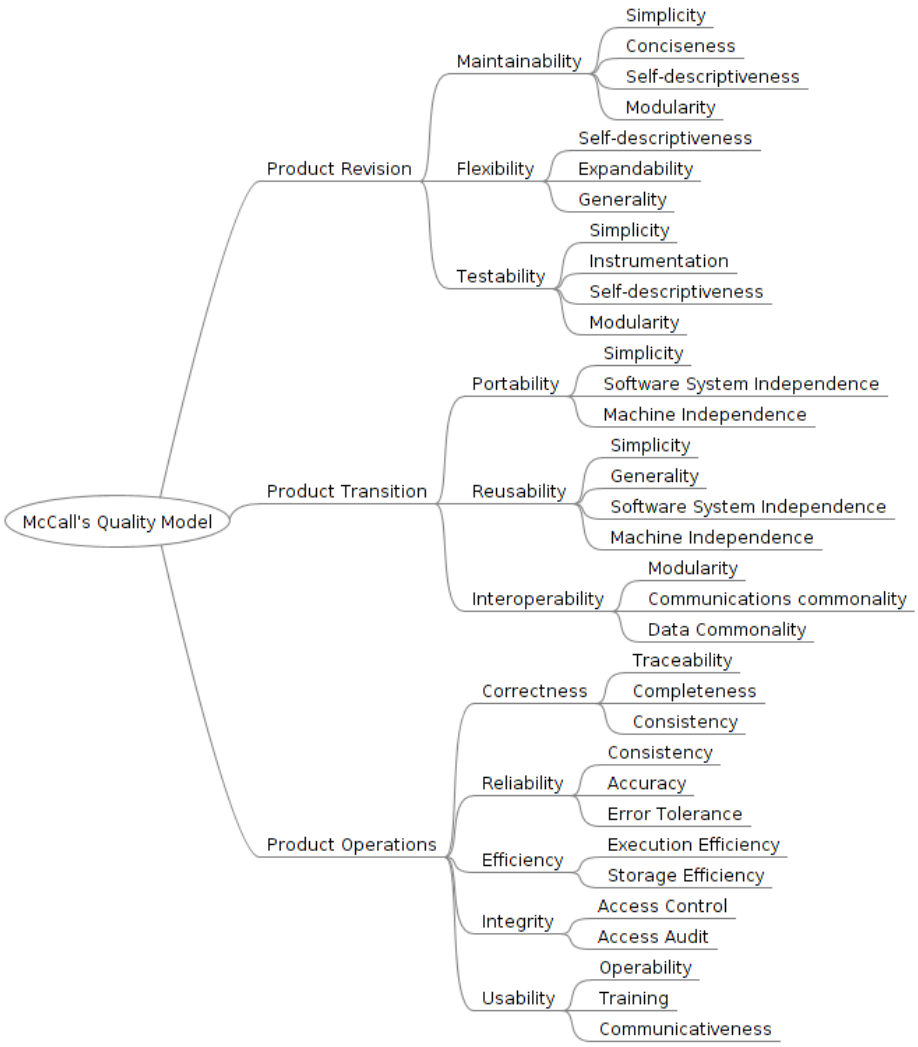


Figure 2. 1: Hierarchy of Major Perspective, Quality Characteristics and Quality Criteria of McCall Quality Model [9]

2.1.2 Boehm's Quality Model (1978)

Barry W. Boehm presented the second quality model in 1978. It consists of a hierarchical model for quality characteristics: high level, intermediate level, and primitive characteristics.

At the highest level of the quality model, Boehm determined three main software requirements: As-is utility, Maintainability, and Portability [10]:

As-is utility:

- Reliability: the degree of software to perform its intended functions satisfactorily.
- Efficiency: ideal use of resources during a correct execution.

- Human Engineering: ease of use, usability.

Maintainability:
- Testability: the ease of validating the software requirements.
- Understandability: the ease of software to be easily comprehended.
- Modifiability: the ability to change the software to meet new requirements.

Portability: the ease of changing the software to accommodate a new environment.

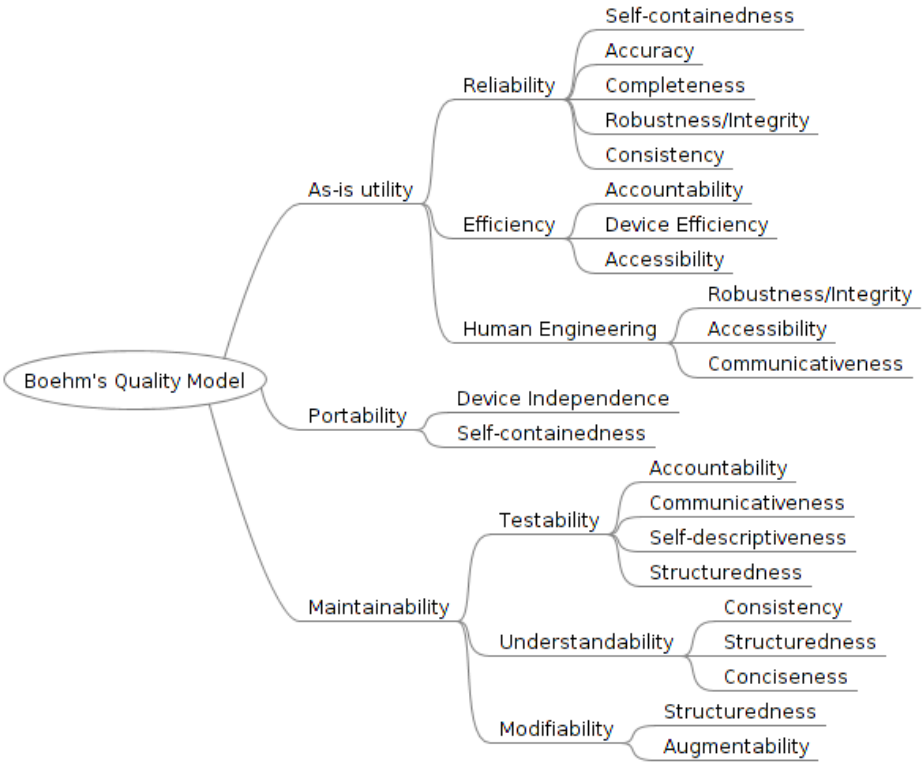


Figure 2. 2: Hierarchy of High Level, Intermediate Level and Primitive Characteristics of Boehm's Quality Model [11]

The hierarchical structure is similar to McCall's quality model, but additionally quality model of Boehm contains hardware performance that is missing in McCall's quality model. [12]

2.1.3 FURPS (1987)

Robert Grady and Hewlett Packard presented FURPS Model in 1987. They built FURPS model for the Rational Software Company. [12] It has two main categories namely functional requirement and non-functional requirement.

Functional Requirements
- Functionality: "includes feature sets, capabilities, and security." [12]

Non-functional Requirements also known as URPS

- Usability: “includes human factors, overall aesthetics, consistency, and documentation.” [12]
- Reliability: “frequency and severity of the failure, recoverability, predictability, accuracy, and mean time between failures (MTBF).” [12]
- Performance: “arranges conditions on functional requirements such as speed, efficiency, availability, accuracy, throughput, response time, recovery time, and resource usage.” [12]
- Supportability: “include testability, extensibility, adaptability, maintainability, compatibility, configurability, serviceability, installability, and localizability.” [12]

FURPS model is also a hierarchical definition model. “Defining quality” is the main purpose of FURPS model. [13] One disadvantage of the FURPS model is that they did not consider software products’ portability as a characteristic [14].

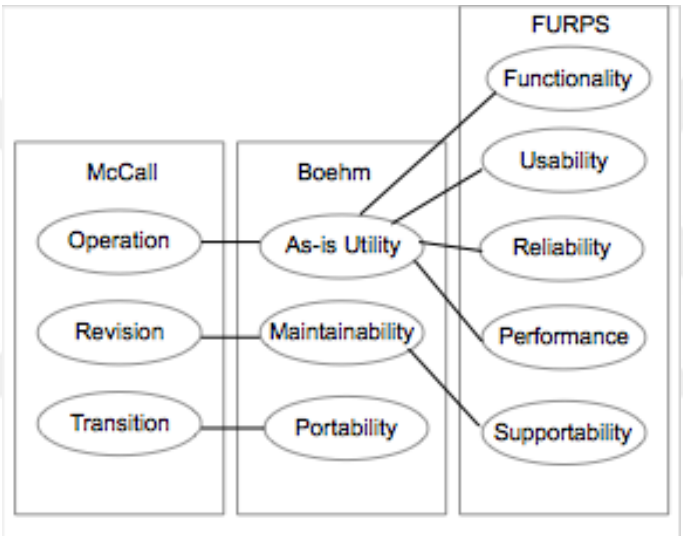


Figure 2. 3: McCall, Boehm, FURPS Models

2.1.4 ISO/IEC 9126 Standard Quality Model (1991)

The International Organization for Standardization (ISO) and The International Electrotechnical Commission (IEC) introduced ISO/IEC 9126, which is an international standard for the evolution of software. The purpose of the standard is defining a quality model and a set of guidelines to measure the software quality characteristics. Hence, the model is divided into four parts: quality model, external metrics, internal metrics and quality in use metrics. ISO/IEC 9126 Part-1 is an extension of McCall, Boehm, and FURPS studies. It decomposes characteristics into two categories namely external characteristics and internal characteristics.

External Characteristics

- Functionality: “the ability of the product to provide functions that meet stated and implied needs.” [15]
- Reliability: “the ability of the product to maintain a specified level of performance.” [15]

- Usability: “the ability of the product to be understood by the user.” [15]
- Efficiency: “the ability of the product to provide appropriate performance, relative to the amount of resources used.” [15]

Internal Characteristics

- Maintainability: “the ability of the product to be modified.” [15]
- Portability: “the ability of the product to be transferred from one environment to another.” [15]

Each of these characteristics has sub-characteristics as illustrated in Figure 2.4.

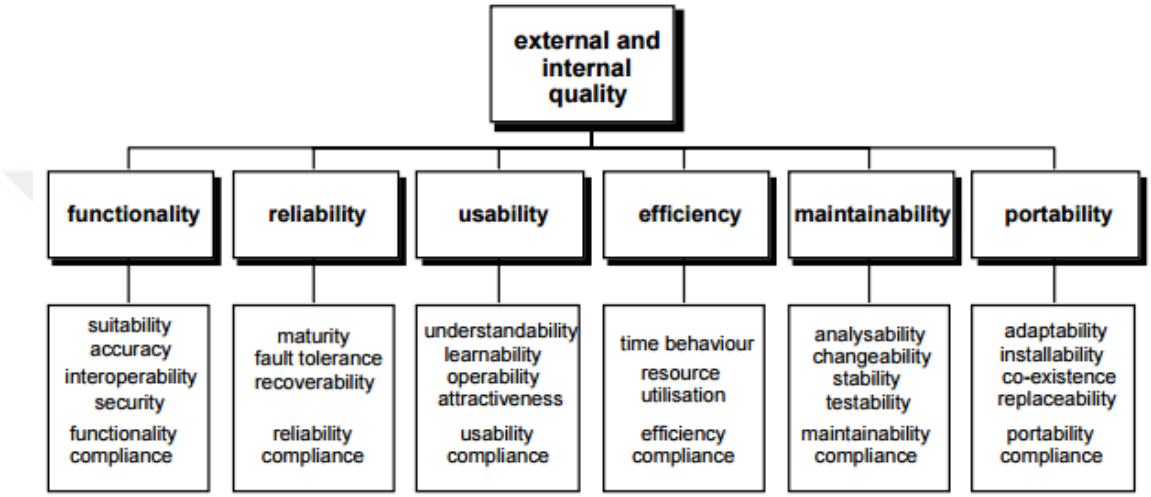


Figure 2. 4: Quality Model of ISO/IEC 9126 [15]

2.1.5 Dromey’s Quality Model (1995)

R. Geoff Dromey presented Dromey’s Quality Model in 1995. The quality model is product based stating that quality evaluation differs for each product. Dromey’s Model focused on relationships between the characteristics and the sub-characteristics of quality. There are three principle elements in this quality model [11]:

1. Product properties that affects the quality.
2. High-level quality characteristics.
3. Linking the quality characteristics with the product properties.

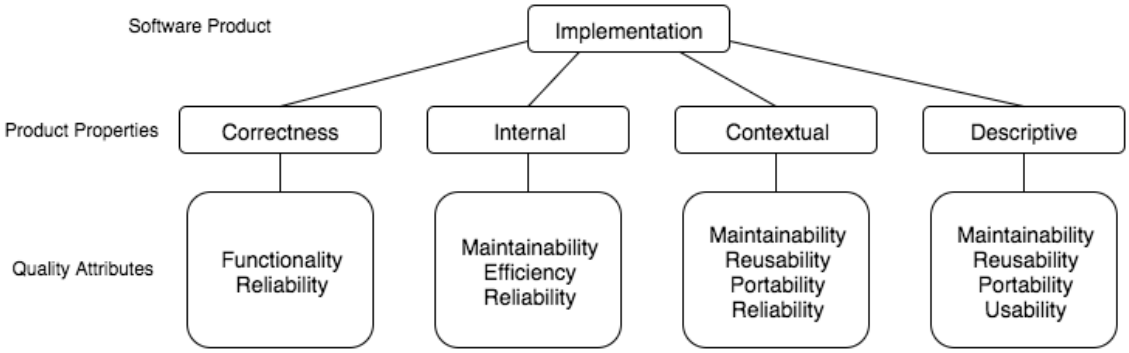


Figure 2. 5: Principles of Dromey's Quality Model [16]

2.1.6 Bansiya's QMOOD Model (2002)

Jagdish Bansiya presented Bansiya's Quality Model in 2002. The model extends Dromey's Model and it is a hierarchical Quality Model for Object Oriented Design (QMOOD). Development of this model includes four levels.

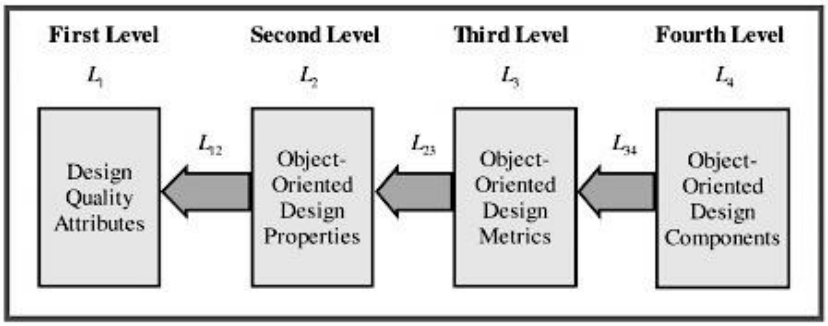


Figure 2. 6: Levels in QMOOD [18]

2.1.6.1 Design Quality Characteristics (L_1)

They examined ISO/IEC 9126 quality characteristics and selected applicable characteristics for design quality. They included two new characteristics that were determined as important for the object-oriented design quality assessment in the model: Reusability and Flexibility. Thus, they identified six Object-Oriented systems characteristics.

- Functionality: "the responsibilities assigned to be classes of design, which are made available by the classes through their public interfaces." [17]
- Effectiveness: "ability to achieve the desired functionality and behavior using OO design concepts and techniques." [17]
- Understandability: "related to the complexity of the design structure." [17]
- Extendibility: "presence and usage of properties in an existing design that allow for the incorporation of new requirements in the design." [17]
- Reusability: "characteristics that allow a design to be reapplied to a new problem without significant effort." [17]

- Flexibility: “the ability of a design to be adapted to provide functionally related capabilities.” [17]

2.1.6.2 Object-Oriented Design Properties (L_2)

They identified design properties as follows [17]: Design Size, Hierarchies, Abstraction, Encapsulation, Coupling, Cohesion, Composition, Inheritance, Polymorphism, Messaging, and Complexity.

2.1.6.3 Object-Oriented Design Metrics (L_3)

They identified design metrics as follows [17]: Design Size in Classes, Number of Hierarchies, Average Number of Ancestors, Data Access Metric, Direct Class Coupling, Cohesion among Methods of Class, Measure of Aggregation, Measure of Functional Abstraction, Number of Polymorphic Methods, Class Interface Size, and Number of Methods.

2.1.6.4 Object-Oriented Design Components (L_4)

They identified design components as follows [17]: objects, classes and the relationships between them.

After these steps, they identified the relationship of quality characteristics and design properties by reviewing object-oriented development books and publications. They developed the final form of QMOOD quality model.

2.1.7 ISO/IEC 25000 Standard (SQuaRE’s Model) (2011)

The purpose of this standard is providing a general overview of Systems and Software Quality Requirements and Evaluation (SQuaRE) contents, common reference models and definitions. ISO/IEC 9126-1:2001 standard has been revised by ISO/IEC 25010:2011. They identified eight quality characteristics:

- Functional Suitability: “degree of a product or system to provide functions that meet stated and implied needs.” [18]
- Performance efficiency: “performance related to the amount of resources used under stated conditions.” [18]
- Compatibility: “degree of a product, system or component to exchange information with other products, systems or components, and perform its required functions, while sharing the same hardware or software environment.” [18]
- Usability: “degree of a product or system to be used by specified users to succeed specified goals with effectiveness, efficiency, and satisfaction.” [18]
- Reliability: “degree of a system, product or component to perform specified functions under specified conditions for a specified period.” [18]
- Security: “degree of a product or system to protect information and data.” [18]

- Maintainability: “degree of effectiveness and efficiency of a product or system to be modified by the intended maintainers.” [18]
- Portability: “degree of effectiveness and efficiency of a system, product or component to be transferred from one environment to another.” [18]

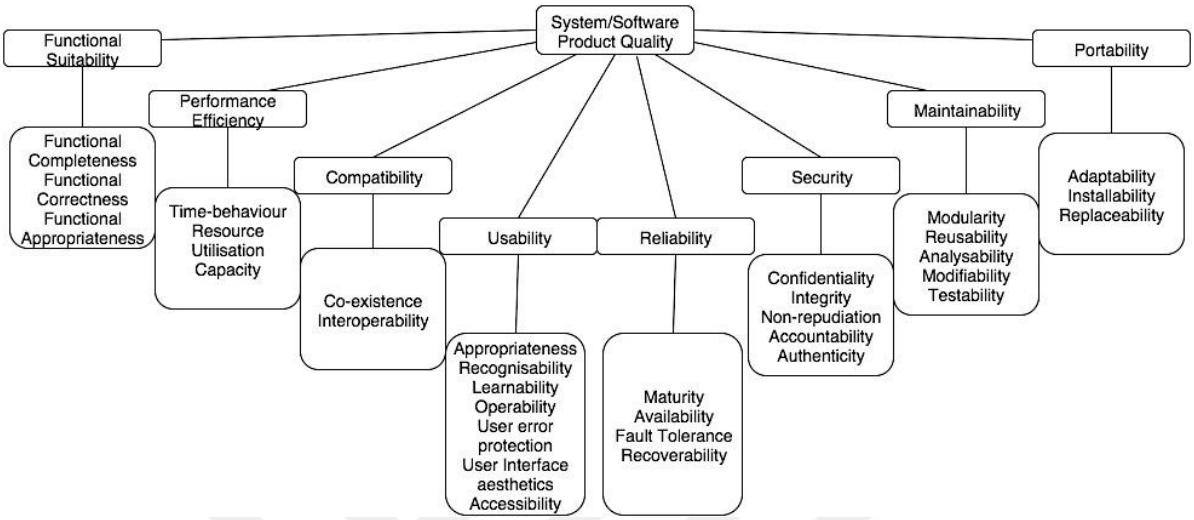


Figure 2. 7: Quality Model of ISO/IEC 25010 [18]

2.2 QUALITY MODELS FOR MOBILE APPLICATIONS

Franke and Kowalewski [19] examined McCall’s, Boehm, and ISO/IEC 9126 software quality models in their study. They extracted quality characteristics from these models and presented a mobile applications quality model as illustrated in Figure 2.8. Afterwards, they applied their model to two Android applications to assess their quality. What is missing in this study is that it only contains the identification of 4 main mobile quality characteristics: Usability, Efficiency, Data Persistence, Flexibility and three sub-characteristics of Flexibility: Adaptability, Portability, Extensibility. The quality model was not validated. How to measure the identified quality characteristics was not considered. Therefore, they did not use metrics for measurement. They selected two non-open source Android applications for case study and evaluated their quality subjectively. Their evaluation was based on their ideas. They did not include an evaluation process, which was statistically supported, in this study.

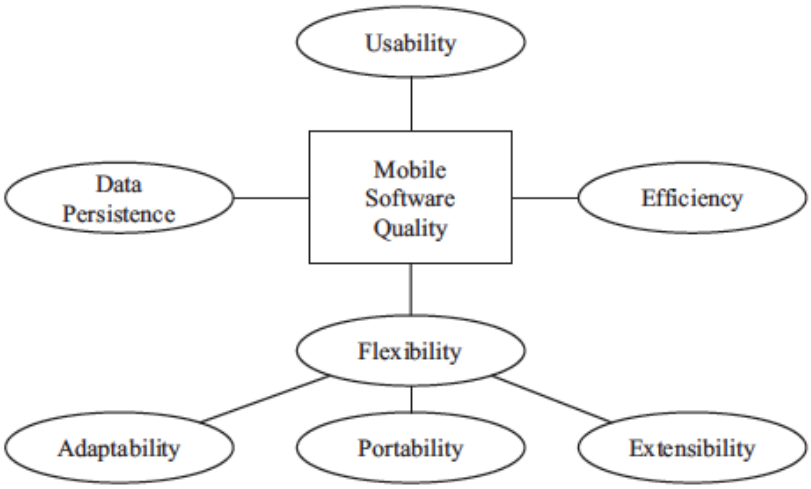


Figure 2. 8: Franke et al. Mobile Applications Quality Model [19]

Franke and Weise [20] provided a framework based on existing quality models, quality metrics, and design patterns for testing mobile applications. They first analyzed statistically the quality of mobile applications. Based on the results they defined methods and tools for testing the lacks, which occurred in the analysis phase. After that, they applied the tools and methods in a case study to assess the provided framework. The most important deficiency in this study is that the results of the case studies are not included. They stated some example metrics that can be used to analyze the source code of software for mobile devices. These examples are Mc-Cabe Cyclomatic Complexity, Weighted Methods per Class and Lack of Cohesion of Methods, but these metrics are not used in the study.

Idri, Moumane, and Abran [21] studied the use of the ISO/IEC 9126 software quality standard to the limitations (e.g. Lower Bandwidth, Frequent Disconnection, and Limited Energy Autonomy) of the mobile environment. They identified the influence of the limitations on quality characteristics. The quality model was useful for assessing Reliability, Usability and Efficiency characteristics. What is missing in this study that it only contains the metrics related to mobile limitations. Besides, it is not a study covering all the quality characteristics; only three of them are covered.

Zahra, Khalid, and Javed [22] presented a mobile application quality model. They extracted quality characteristics of the model from ISO / IEC 9126 standard as illustrated in Figure 10. This study only contains the identification of mobile quality characteristics. Identification of metrics for mobile-specific measurements was not considered. In addition, they did not apply their model to mobile applications to show the applicability of the model.

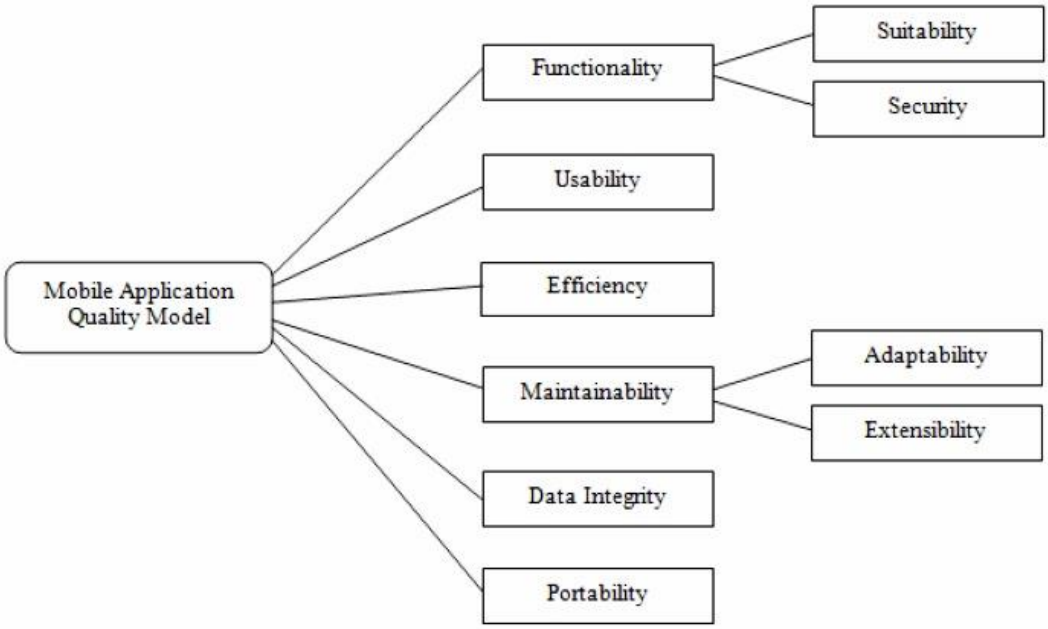


Figure 2. 9: Zahra et al. Mobile Applications Quality Model [22]

Jost, Huber and Hericko [23] examined whether the traditional software metrics are suitable for assessing the source code of mobile applications. They developed a small-scale application for three different platforms (Android, IOS, Windows Phone). They evaluated the source codes of the applications by using traditional software metrics. Because the results of the analysis were slightly different between platforms, the hypothesis "traditional software metrics can be used for mobile applications' source codes" was rejected. Mobile application quality model development and mobile-specific metric determination are missing in this study. They used traditional software metrics.

Hecht, Benomar, Rouvoy, Moha and Duchien [24] introduced the PAPRIKA tool to monitor the evolution of mobile apps quality based on antipatterns. The antipattern detection was based on software metrics computed by the tool. They considered three Object Oriented antipatterns, which are Blob Class, Long Method, Complex Class, and four Android antipatterns, which are Member Ignoring Method, Leaking Inner Class, UI Overdraw, and Heavy Broadcast Receiver. They utilized the detected antipatterns in the evaluation of mobile applications quality. Development of a mobile application quality model is missing in this study. They identified the antipatterns and applied them in mobile applications in a case study to evaluate software quality score of mobile applications.

CHAPTER III

3 ANDROID QUALITY MODEL DEVELOPMENT

To quantify quality, researchers have developed quality models as explained in Chapter 2. All of these models share a common difficulty: they are less applicable for Android Applications. Our model resolves this difficulty. In this chapter, we explained the methodology used in the development of the proposed Android Quality Model in a detailed manner. The first section contains the information about the identification of quality characteristics. The second section contains the information about the identification of metrics specific for Android applications. The third section contains the information about determining the relationship between the identified quality characteristics and the identified metrics. Finally in the last section, we present the developed Android Applications Quality Model. We illustrated research methodology of the study in Figure 3.1.

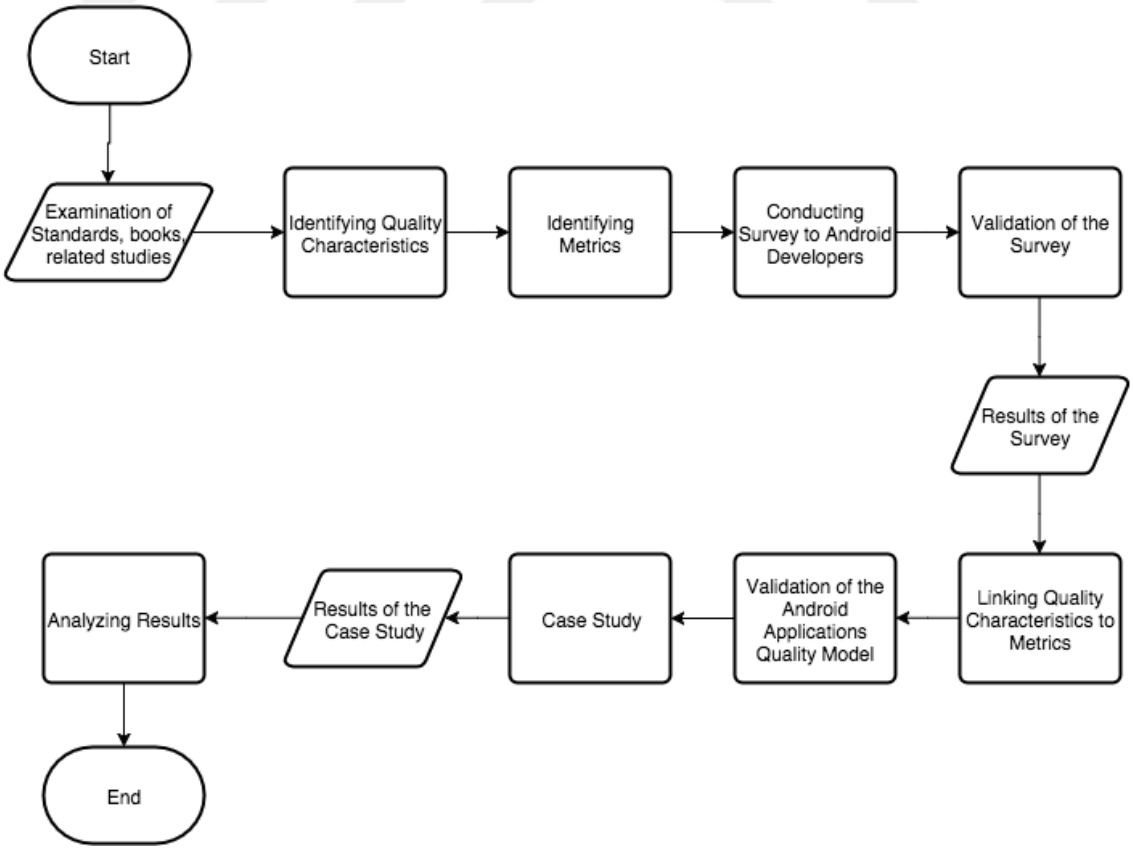


Figure 3. 1: The Research Methodology

3.1 IDENTIFYING QUALITY CHARACTERISTICS

We considered ISO/IEC SQuaRE Quality Standard, Object-Oriented models, and components that are specific to Android Applications while identifying quality characteristics. We selected the quality characteristics "Functional Suitability", "Reliability", "Performance Efficiency", "Portability", "Maintainability", "Usability", "Security" and "Compatibility" from ISO/IEC 25010 Quality Standard as the initial set of quality characteristics. In addition to these quality characteristics, we selected "Data Integrity" as a characteristic considering the studies in this field [19] [20] [22]. We provided the definitions of the quality characteristics identified for our Android Applications Quality Model and reasons of the identification of these characteristics below. We illustrated the Android Applications Quality Model in Figure 3.2.

A. Functional Suitability

Functional Suitability is the degree of a product/system to provide functions that fulfill needs [18]. Functional suitability for mobile applications means an application meets the needs of a particular user. If there is too much extra functionality in an application, it devastates the user. On the other hand, if functionalities are less than it should be needs of the user are not fulfilled. It is expected from a mobile application to answer these three main questions [26]:

- Are the application features included?
- Are the application features working properly?
- Is the install/uninstall process of the application working properly?

B. Reliability

Reliability is the degree of a product/system/component to perform specified functions under stated conditions for a specified period [18]. Reliability means the stability of mobile applications. Users do not want to deal with an application with full of bugs that causes a frustration overload. Besides, users do not want to deal with a slow application that causes waiting a long time to load the application. A mobile application with poor stability leads quickly to abandonment. According to research from TechCrunch, "users have low tolerance for buggy apps, only 16% will try a failing app more than twice" [27].

C. Performance Efficiency

Performance Efficiency is related to the amount of resources used under specified conditions [18]. Mobile resources are limited; thus, they should be used optimally. Battery power and phone memory are limited resources on the mobile phones. Users typically only have $1/1000^{th}$ as much memory in a mobile phone as they have on a desktop computer [28]. Besides mobile phones do not have a limitless supply of electrical power. Users expect the battery in their mobile devices to last at least eight hours. To meet this requirement, both the hardware and the software on a mobile

device should be power efficient. Users tend to uninstall apps that run their battery down quickly, and they write unfavorable reviews about these applications [29].

D. Portability

Portability is the degree of efficiency and effectiveness of a product/system/component to be transferred from one environment to another [18]. Portability measures if an application is able to run on different devices. The mobile market is growing rapidly and there are a wide range of models for each mobile device. "The worldwide smartphone market grew 13.0% year over year in 2015 Q2, with 341.5 million shipments" according to the data from the International Data Corporation (IDC) Worldwide Quarterly Mobile Phone Tracker. Android dominated the market of smartphones with a share of 82.8% [30]. Samsung and other players such as Huawei, Xiaomi, ZTE, LG, Nexus, Sony, Asus contribute this percentage [31]. If the diversity in the market is taken into consideration, users should be able to run the same application on different mobile devices with the same efficiency.

E. Maintainability

Maintainability is the degree of efficiency and effectiveness of a product/system to be modified by the maintainers [18]. Mobile Application Markets are already big, and they are going to keep getting bigger. According to the statistics, the number of applications available for download in Google Play Store as of July 2015 is 1.6 million. Apple's App Store is the second largest application store with 1.5 million available applications [4]. Thus, markets are rapidly changing. Developers should update mobile applications according to the changing requirements. Maintainability is helpful at this point. Extensibility is the sub-characteristic of maintainability. Mobile applications have to be extensible for new hardware components, or they can be extended if new releases of applications available in App Stores.

F. Usability

Usability is the degree of a product/system to be used by specified users to succeed specified goals with efficiency, effectiveness, and satisfaction [18]. PACMAD Usability Model identifies seven attributes for mobile applications [32]:

1. Efficiency is the ease of a user to complete a transaction with speed and accuracy.
2. Effectiveness is the ease of a user to complete a transaction in a stated context.
3. Satisfaction is the level of pleasantness of the user while using the application.
4. Memorability is the ability of a user to remember how to use an application effectively.
5. Learnability is the ability of a user to gain proficiency with an application.
6. Cognitive Load means the amount of cognitive processing necessary for a user to use the application.
7. Errors attribute express how well the user can complete the transactions without errors.

Usability is related to improve user experience and response about mobile applications. First seconds of mobile applications are critical because users decide to use the application or abandon the application [22].

G. Security

Security is the degree of a product/system to protect information and data [18]. Mobile application number in App Stores is increasing dramatically. The question is: Are mobile applications secure enough? According to the recent research by Arxan, the rates of the hacked applications are as follows [33]:

- 100% of the top 100 paid apps on the Google Android platform
- 56% of the top 100 paid apps for Apple iOS
- 73% of popular free apps on Android
- 53% of popular free apps on Apple iOS

These numbers obviously present the necessity of security for mobile applications.

H. Compatibility

Compatibility is the degree of a product/system/component to exchange information with other products/systems/components, and performs its required functions while sharing the same environment [18]. Mobile applications may also exchange information with each other by using Bound Services, which are implementation of Service classes. Bound Service allows other applications to bind to it and interact with it. Inter-process communication (IPC) is performed [34].

İ. Data Integrity

Data Integrity is the ability of a mobile application to keep information when state changes happen [18]. Data integrity is the main component of information security. It provides data, which is stored in a database, data warehouse, data mart or another construct, to be kept accurately and consistently. Data should be kept free from corruption or modification. When mobile applications are paused, killed or a call arrived, the current state of running applications should be saved [22]. Data corruption, a form of data loss leads users to abandonment and developers to failure.

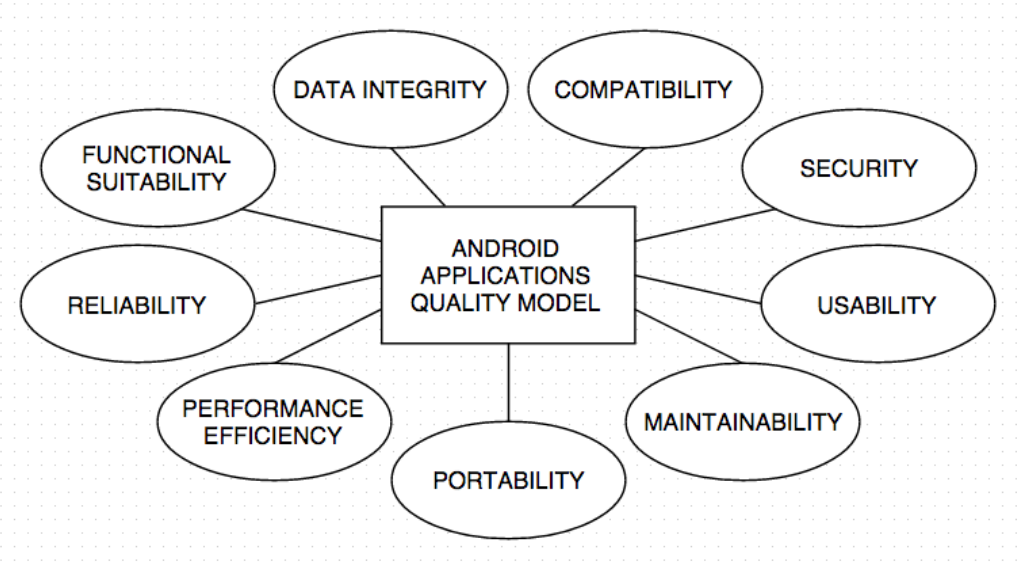


Figure 3. 2: The Android Applications Quality Model

3.2 IDENTIFYING METRICS

Source code metrics are commonly used in assessing the quality of software systems. They present an objective way to get concrete information about the source code. Source code metrics for quality assessment of Object-Oriented Design was identified in QMOOD by Bansiya[17]. In a study of Jetter from the University of Zurich, Bansiya's Quality Model was adapted for the Java programming language. They identified source code metrics related to the Java source code [35]. Metrics that were used in these two studies is given in Table 3.1.

We analyzed related studies, quality standards and extracted metrics applicable for Android Applications from these sources. Source code metrics like Number of Classes, Depth of Inheritance Tree, Instability, Lack of Cohesion of Methods, Number of Methods and McCabe Cyclomatic Complexity can also be applicable to Android applications' source codes.

Table 3. 1: METRICS USED FOR QUALITY ASSESSMENT FOR OBJECT-ORIENTED DESIGN IN LITERATURE

Metrics used by Bansiya [17]	**Metrics used by Jetter [35]**
Design Size in Classes	Number of Classes
Number of Hierarchies	Depth of Inheritance Tree
Average Number of Ancestors	Abstractness
Data Access Metric	-
Direct Class Coupling	Instability
Cohesion Among Methods of Classes	1 / Lack of Cohesion of Methods
Measure of Aggregation	Number of Attributes
Measure of Functional Abstraction	$1 - \frac{\text{Number of Overridden Methods}}{\text{Number of Methods}}$
Number of Polymorphic Methods	Number of Overridden Methods

Class Interface Size	Number of Methods
Number of Methods	Weighted Methods per Class

Android development is Java-Based and Object-Oriented, but there are key differences. Java programs have a main function On the contrary; Android applications do not have a main function. They have onCreate, onResume, onPause and onDestroy functions. Developers overwrite these functions while developing Android applications. [36]. Due to the difference in source code structures, we need an identification of metrics that are specific for Android application. By reviewing Android Programming books, tutorials and publications extensively [25] [36] [37] [38] [39] [40] [41] [42] [43] [44], we identified eight new Android-specific metrics: Number of Activities, Number of Services, Number of Broadcast Receivers, Number of Content Providers, Minimum SDK Version, Target SDK Version, Number of Intents and Number of Permission. These components are special for Android development. The definitions of these Android components [25] are given below:

- Activity: Provides a screen to users to interact to do a transaction such as: dialing the phone, taking a photo, sending an email, and viewing a map.

- Broadcast Receiver: Responds to broadcast announcements from the system or from other applications. Broadcast examples from the system are notifying that the battery is low, a picture was captured, or the screen has turned off.

- Content Provider: Manages access to a structured set of data. Content provider examples are Contacts, which allows other applications to access user information, Media Store, which allows other applications to access or store media files.

- Intent: Allows application components to request functionality from other Android components. The intent is a message to communicate an action such as View Video, Play Game.

- SDK Version: An integer value designating the API Level required for an application to run. For example, API Level of Android 6.0 Platform is 23, API Level of Android 3.0.x is 11 and API Level of Android 1.0 is 1.

- Permission: Request permissions that applications must be granted to operate correctly. Users confirm permissions when they install the application.

- Service: Performs long-running operations in the background. Services do not provide a user interface in contrast to the activities.

In addition to the Android-specific metrics, we identified three metrics that may be applicable to Android applications: Number of Dialogs, Number of Threads, and Number of Tables in Database. These metrics are not Android-specific. They are also applicable to Object-Oriented Programs. We identified them because we hypothesize that they might affect Android applications' quality.

The resulting list of metrics and descriptions of them are given in Table 3.2.

Table 3. 2: LIST OF METRICS USED IN OUR MODEL

METRIC	**TYPE**	**DESCRIPTION**
Num of Classes	OO	This metric is the count of the total number of classes in a source code.
Depth of Inheritance Tree	OO	This metric calculates the maximum length of a path from a class to the root class in the inheritance hierarchy.
Instability	OO	This metric is the ratio of efferent coupling to total coupling (Efferent + Afferent). Instability shows the package's resilience to change. [46]. - *Afferent Coupling: "The number of classes in other packages that depend upon classes within the package is an indicator of the package's responsibility. Afferent = incoming" [46].* - *Efferent Coupling: "The number of classes in other packages that the classes in the package depend upon is an indicator of the package's dependence on externalities. Efferent = outgoing" [46].*
Lack of Cohesion of Methods	OO	This metric measures the relation degree of methods and fields to each other within a class. It measures the cohesion of a class.
Num of Attributes	OO	This metric is the count of the total number of attributes in source code.
Num of Methods	OO	This metric is the count of the total number of methods in source code.
McCabe Cyclomatic Complexity	OO	This metric measures the number of linearly independent paths through source code of a program.
Num of Activities	Android	This metric is the count of the total number of classes that extend Activities in a source code.
Num of Services	Android	This metric is the count of the total number of classes that extend Services in a source code.
Num of BroadcastReceivers	Android	This metric is the count of the total

		number of classes that extend BroadcastReceivers in a source code.
Num of ContentProviders	Android	This metric is the count of the total number of classes that extend ContentProviders in a source code.
Num of Dialogs	OO Android	This metric is the count of the total number of dialogs in source code. Dialogs are small windows that ask users to enter information or make decisions.
Num of Threads	OO Android	This metric is the count of the total number of classes that extend Threads in a source code.
Android:minSdkVersion	Android	This metric is an integer, which indicates the minimum API Level required the application to run.
Android:targetSdkVersion	Android	This metric is an integer, which indicates the target API Level required the application to run.
Num of Intents	Android	This metric is the count of the total number of Intents in a source code.
Num of Tables in Database	OO Android	This metric is the count of the total number of tables in the database of the application.
Num of <uses-permission>	Android	This metric is the number of requested permissions that the application should have to operate correctly.

3.3 METRICS - QUALITY CHARACTERISTICS RELATIONSHIP

To determine the relationship between the source code metrics and our quality characteristics, we conducted a survey for Android Developers. The survey consists of two parts. The first part includes demographic questions about respondents; second part includes questions to identify the relationships between metrics and quality characteristics. The demographics part includes three questions about years of work experience, the number of developed Android projects and purpose of Android application development. The second part includes 18 questions that expect respondents to select the degree of effect for each metrics on the quality characteristics (Likert Scale in [-1, +1] range). The options are Strongly Negative (-1), Somewhat Negative (-0,5), No Effect (0), Somewhat Positive (+0,5) and Strongly Positive (+1). The survey is prepared and sent as an online survey. The Survey tool of Google Forms [47] was used for the preparation of the survey and acquiring responses from participants. We included the survey and ethics approval form of survey in the Appendix.

Since the survey asks questions about the structure of Android applications, our respondents should have the ability to develop Android Applications. We emailed our survey to Android Developer Groups in Social Media (Facebook, Linkedin),

several IT Companies in Turkey and our personal connections. The mail contains a brief explanation about the study, the purpose of the study and the link to our online survey. 33 Android developers participated the survey. The demographic data of respondents are as follows:

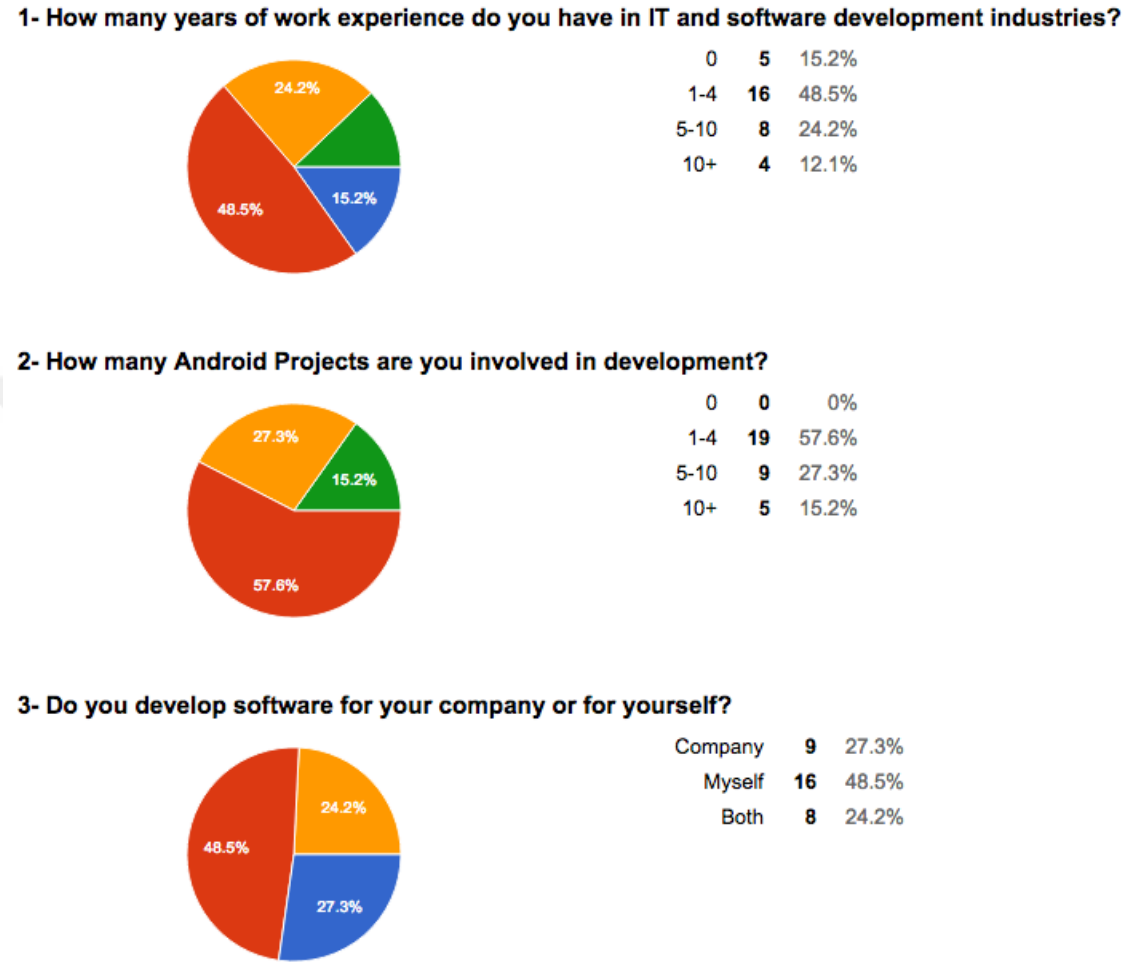


Figure 3. 3: Demographic Data of Respondents

In this study, we employ a quantitative approach to analyzing the data. We used Cronbach's Alpha [48], which is the most common statistic, to investigate the internal consistency (reliability) of surveys. We calculated the Cronbach's Alpha value of 33 responses by using IBM SPSS Statistics [50]. The Alpha values for each question are provided in the Appendix. The average value of all survey questions is 0.82, which indicates the reliability of our survey is in an acceptable range.

3.3.1 Weighting The Relationship

Using the results of the survey, we created a matrix that shows the relationship of metrics and quality characteristics. This matrix is given in Table 3.3. The matrix shows how an increase in a metric given in a row affects the quality characteristics. In this table 0 means no effect, -1 means strong negative effect, -0.5 means negative effect, +0.5 means positive effect, +1 means a strong positive effect on a

characteristic. To determine the effect of a metric, we use the median [53] of responses on the survey for each quality characteristic.

Table 3. 3: METRICS - QUALITY CHARACTERISTICS RELATIONSHIP (THE MEDIAN OF RESPONSES)

Increase in	Functional Suitability	Reliability	Performance Efficiency	Portability	Maintainability	Data Integrity	Usability	Security	Compatibility
# Of Classes (NOC)	+0.5	0	+1.0	+1.0	+0.5	0	+0.5	+0.5	0
Depth of Inheritance Tree (DIT)	+0.5	-0.5	-1.0	+0.5	+0.5	+0.5	0	0	+0.5
Instability (COP)	-0.5	-0.5	0	0	-0.5	+0.5	0	+0.5	+0.5
1 / Lack of Cohesion of Methods (COH)	+0.5	+0.5	0	0	0	+0.5	0	+0.5	+0.5
# Of Attributes (NOAT)	0	+0.5	-0.5	0	0	+0.5	+0.5	+0.5	+0.5
# Of Methods (NOM)	+0.5	0	0	+0.5	-0.5	+0.5	+0.5	+0.5	0
McCabe Cyclomatic Complexity (CYC)	0	0	-1.0	0	0	0	0	0	0
# Of Activities (NOAC)	+0.5	0	0	0	+0.5	+1.0	+0.5	0	0
# Of Services (NOS)	+0.5	+0.5	-0.5	+0.5	+0.5	+0.5	+0.5	-0.5	0
# Of Broadcast Receivers (NOBR)	+0.5	+0.5	0	0	0	0	0	0	0
# Of Content Providers (NOCP)	-0.5	+0.5	+0.5	+0.5	+0.5	0	+0.5	+0.5	0
# Of Dialogs (NOD)	0	0	-1.0	0	0	0	+0.5	0	0
# Of Threads (NOT)	+0.5	+0.5	+1.0	0	+0.5	+0.5	+0.5	0	0
android:minSdkVersion (MSDK)	+0.5	0	0	-1.0	0	0	0	0	0
android:targetSdkVersion (TSDK)	+0.5	0	0	0	0	0	0	0	0
# Of Intents (NOI)	+0.5	0	0	+0.5	0	+0.5	+0.5	+0.5	+1.0
# Of Tables in Database (NOTD)	+0.5	0	-1.0	+0.5	+0.5	+0.5	0	-1.0	+0.5
# Of <uses-permission> (NOP)	0	0	+0.5	0	0	0	-0.5	+0.5	0

By using this matrix, we obtained computation formulas for each quality characteristics. We weighted proportionally the effects of metrics on quality characteristics so that the computed values of all quality characteristics have the same range. We selected ± 1 as a range for the effects of metrics. For this reason, we changed the weights of each metric to ensure that the sum of the new weight values of all metrics equal to ± 1. We used Bansiya's QMOOD study while determining the weights [17]. The resulting computational formulas for quality characteristics are shown in Table 3.4.

Table 3. 4: COMPUTATION FORMULAS FOR QUALITY CHARACTERISTICS

QUALITY CHARACTERISTIC	**COMPUTATION EQUATION**
Functional Suitability	+0.1*(NOC) +0.1*(DIT) -0.1*(COP) +0.1*(COH) +0.1*(NOM) +0.1*(NOAC) +0.1*(NOS) +0.1*(NOBR) -0.1*(NOCP) +0.1*(NOT) +0.1*(MSDK) +0.1*(TSDK) +0.1*(NOI) +0.1*(NOTD)
Reliability	-0.25*(DIT) -0.25*(COP) +0.25*(COH) +0.25*(NOAT) +0.25*(NOS) +0.25*(NOBR) +0.25*(NOCP) +0.25*(NOT)
Performance Efficiency	+0.5*(NOC) -0.5*(DIT) -0.25*(NOAT) -0.5*(CYC) -0.25*(NOS) +0.25*(NOCP) -0.5*(NOD) +0.5(NOT) -0.5(NOTD) +0.25*(NOP)
Portability	+0.33*(NOC) +0.165*(DIT) +0.165*(NOM) +0.165*(NOS) +0.165*(NOCP) -0.33*(MSDK) +0.165*(NOI) +0.165*(NOTD)
Maintainability	+0.2*(NOC) +0.2*(DIT) -0.2*(COP) -0.2*(NOM) +0.2*(NOAC) +0.2*(NOS) +0.2*(NOCP) +0.2*(NOT) +0.2*(NOTD)
Data Integrity	+0.09*(DIT) +0.09*(COP) +0.09*(COH) +0.09*(NOAT) +0.09*(NOM) +0.18*(NOAC) +0.09*(NOS) +0.09*(NOT) +0.09*(NOI) +0.09*(NOTD)
Usability	+0.125*(NOC) +0.125*(NOAT) +0.125*(NOM) +0.125*(NOAC) +0.125*(NOS) +0.125*(NOCP) +0.125*(NOD) +0.125*(NOT) +0.125*(NOI) -0.125*(NOP)
Security	+0.2*(NOC) +0.2*(COP) +0.2*(COH) +0.2*(NOAT) +0.2*(NOM) -0.2*(NOS) +0.2*(NOCP) +0.2*(NOI) -0.4*(NOTD) +0.2*(NOP)
Compatibility	+0.14*(DIT) +0.14*(COP) +0.14*(COH) +0.14*(NOAT) +0.28*(NOI) +0.14*(NOTD)

We provided the minimum and maximum values of the quality characteristics according to the computation formulas in Table 3.5. We took zero (0) as a minimum value of metrics, one (1) as a maximum value of metrics while calculating the minimum and maximum values of the quality characteristics. We created the table to use it while adjusting the ranges of quality characteristics in Section 4.1 and Section 4.5.

Table 3. 5: MIN AND MAX VALUES OF QUALITY CHARACTERISTICS

QUALITY CHARACTERISTIC	**MIN VALUE**	**MAX VALUE**	**TOTAL VALUE**
Functional Suitability	-0.2	+1.2	+1
Reliability	-0,5	+1,5	+1
Performance Efficiency	-2,5	+1,5	-1
Portability	-0,33	+1,32	+0.99
Maintainability	-0,4	+1,4	+1
Data Integrity	0	+0,99	+0,99
Usability	-0,125	+1,125	+1
Security	-0,6	+1,6	+1
Compatibility	0	0,98	+0,98

CHAPTER IV

4 MODEL EVALUATION & RESULTS

In this chapter, we present a validation and the evaluation of the quality model and results of the case studies. The first section contains the validation of the Android Applications Quality Model and determination of a rating scale for quality characteristics. The second section contains the information about selecting Android applications for a case study. The third section contains the information about gathering and normalizing metric data. The results are analyzed in fourth section.

4.1 MODEL VALIDATION

In order to verify the computed values of the Android Application Quality Model, we used a group of three independent evaluators to study the quality of two applications. All evaluators had four to five years of experience in software development and had knowledge of the Android application programming.

To select an open source application as a validation suite, we used an application called F-Droid [51], which is a catalogue of free and open-source applications for Android platform. We searched for application based on the following criteria from F-Droid's list:

- To be popular
- To be open source
- To be small (max 5000 KLOC)
- To have at least two releases.

We selected two releases of 2048-android application as a validation suite: the first release and the last release. The 2048-android application [49] is a simple puzzle game, which consists of 4.275 KLOC. It provides the expected criteria.

Three independent evaluators analyzed the source codes of the 2048-android v1 and 2048-android v2.06 and scored the quality characteristics. Participants scored each quality characteristics on a range of [0, 10]. Table 4.1 shows the scores of 3 evaluators and the average values of quality characteristics.

Table 4. 1: SCORES OF THE EVALUATORS TO 2048-ANDROID APPLICATIONS

	2048-android v1.0				2048-android v2.06			
Evaluators	E1	E2	E3	AVG	E1	E2	E3	AVG
Functional Suitability	5	4	6	5	8	7	10	8,33
Reliability	4	3	6	4,33	7	9	10	8,66
Performance Efficiency	4	2	5	3,66	6	5	8	6,33
Portability	2	5	7	4,66	8	10	8	8,66
Maintainability	7	6	8	7	4	6	7	5,66
Data Integrity	5	3	6	4,66	8	7	9	8
Usability	4	5	8	5,66	7	9	10	8,66
Security	6	4	7	5,66	8	7	8	7,66
Compatibility	2	3	5	3,33	6	5	5	5,33

After that, to validate the Android Application Quality Model we calculated the quality scores for the two releases of the selected application by analyzing source codes of them. To calculate the quality scores, we first gathered the metrics of the two releases. We described the methods we used for gathering metrics in Section 4.3. The metric values for the releases of the 2048-android application are stated in Table 4.2. Since we combined actual metric values of different ranges in the computation of the quality characteristics, normalization of these metric values was necessary. Thus, we normalized these metric values. The calculation and normalization methods that were used in this study are explained in Section 4.3.

Table 4. 2: METRIC VALUES FOR 2048-ANDROID

	2048-android v1.0	2048-android v2.06
# Of Classes (NOC)	11	12
Depth of Inheritance Tree (DIT)	5	5
Instability (COP)	0.5	1
1 / Lack of Cohesion of Methods (COH)	0	16.1
# Of Attributes (NOAT)	0	4
# Of Methods (NOM)	2	8
McCabe Cyclomatic Complexity (CYC)	1	2.125
# Of Activities (NOAC)	1	1
# Of Services (NOS)	0	0
# Of Broadcast Receivers (NOBR)	0	0
# Of Content Providers (NOCP)	0	0
# Of Dialogs (NOD)	0	0
# Of Threads (NOT)	0	0
android:minSdkVersion (MSDK)	10	8
android:targetSdkVersion (TSDK)	19	23
# Of Intents (NOI)	0	0
# Of Tables in Database (NOTD)	0	0
# Of <uses-permission> (NOP)	0	0

We calculated the values of identified nine quality characteristics by using the computation formulas stated in Table 3.4. We used normalized values of the metrics for each of the formula. Table 4.3 shows the computed values of nine quality characteristics for the two releases of the 2048-android game.

Table 4. 3: COMPUTED QUALITY CHARACTERISTICS FOR 2048-ANDROID

	2048-android v1.0	2048-android v2.06
Functional Suitability	0,1	0,3
Reliability	0	0,25
Performance Efficiency	0	-0,25
Portability	-0,33	0,495
Maintainability	0	-0,2
Data Integrity	0	0,36
Usability	0	0,375
Security	0	1
Compatibility	0	0,42

Afterwards, to compare the results of the quality model and the results of the evaluators we changed the ranges of computed quality characteristics to [0, 10] by using the formula provided below [52]:

$$NewValue = \frac{(Value - Min) * (NewMax - NewMin)}{Max - Min} + NewMin$$

Where

- Min and Max: The minimum and maximum values of quality characteristics provided in Table 3.5.
- NewMin: New minimum value is 0
- NewMax: New maximum value is 10
- Value: Values of quality characteristics provided in Table 4.3.
- NewValue: New values of quality characteristics within the new range [0, 10].

We calculated new values of quality characteristics. We provided them in Table 4.4.

Table 4. 4: COMPUTED QUALITY CHARACTERISTICS FOR 2048-ANDROID WITHIN THE NEW RANGE [0, 10]

	2048-android v1.0	2048-android v2.06
Functional Suitability	2,14	3,57
Reliability	2,5	3,75
Performance Efficiency	3,125	5,625
Portability	0	5
Maintainability	5,7	1,11
Data Integrity	0	3,63
Usability	1	4
Security	2,72	7,27

Compatibility	0	4,28

We used Paired Samples T-Test to calculate the difference between results of the quality model and scores of the evaluators. If there are two data sets in which observations in one data set can be paired with observations in the other data set, generally paired samples t-test is used for comparison in statistics [54]. Thus, we selected this test to compare our two data sets.

We obtained the Sig (2-Tailed) values and t values of 9 pairs by using IBM SPSS Statistics [50]. We provided the values in Figure 4.1 and Figure 4.2.

Paired Samples Test

		Paired Differences			
					95% Confidence ...
		Mean	Std. Deviation	Std. Error Mean	Lower
Pair 1	FunctionalSuitabilityEvaluator - FunctionalSuitabilityModel	3,81000	1,34350	,95000	-8,26089
Pair 2	ReliabilityEvaluator - ReliabilityModel	4,20500	,99702	,70500	-4,75287
Pair 3	PerformanceEfficiencyEvaluator - PerformanceEfficiencyModel	,62000	,12021	,08500	-,46003
Pair 4	PortabilityEvaluator - PortabilityModel	4,16000	,70711	,50000	-2,19310
Pair 5	MaintainabilityEvaluator - MaintainabilityModel	2,92500	2,29810	1,62500	-17,72258
Pair 6	DataIntegrityEvaluator - DataIntegrityModel	4,51500	,20506	,14500	2,67260
Pair 8	SecurityEvaluator - SecurityModel	1,66500	1,80312	1,27500	-14,53541
Pair 9	CompatibilityEvaluator - CompatibilityModel	2,19000	1,61220	1,14000	-12,29507

Figure 4. 1: Paired Samples T-Test of the Data Sets (Scores of Evaluators & Results of the Model)

		Paired ...			
		95% Confidence Interval of the ...			
		Upper	t	df	Sig. (2-tailed)
Pair 1	FunctionalSuitabilityEvaluator - FunctionalSuitabilityModel	15,88089	4,011	1	,156
Pair 2	ReliabilityEvaluator - ReliabilityModel	13,16287	5,965	1	,106
Pair 3	PerformanceEfficiencyEvaluator - PerformanceEfficiencyModel	1,70003	7,294	1	,087
Pair 4	PortabilityEvaluator - PortabilityModel	10,51310	8,320	1	,076
Pair 5	MaintainabilityEvaluator - MaintainabilityModel	23,57258	1,800	1	,323
Pair 6	DataIntegrityEvaluator - DataIntegrityModel	6,35740	31,138	1	,020
Pair 8	SecurityEvaluator - SecurityModel	17,86541	1,306	1	,416
Pair 9	CompatibilityEvaluator - CompatibilityModel	16,67507	1,921	1	,306

Figure 4. 2: Paired Samples T-Test of the Data Sets (Scores of Evaluators & Results of the Model)

All Sig (2-Tailed) values of the pairs are greater than 0.005. Thus, according to paired samples t-test results, there is a statistically significant correlation between the two datasets. This correlation validates our Android Applications Quality Model and supports the reliability of the model.

4.1.1 Determination of a Rating Scale

In order to elicit information about the results, we determined a rating for the quality scores of quality characteristics of Android applications. The rating scale contains six categories: 0-1 Very Poor, 1-3 Poor, 3-5 Fair, 5-7 Good, 7-9 Very Good, and 9-10 Excellent.

Very Poor	Poor	Fair	Good	Very Good	Excellent
0 1	3	5	7	9	10

We applied our rating scale to the results of quality characteristics presented in Table 4.4 to figure out which categories they enter. According to the rating scale, the quality characteristics computed by the Android Applications Quality Model for 2048-Android applications enter into the categories provided in Table 4.5.

Table 4. 5: THE CATEGORIES OF QUALITY CHARACTERISTICS OF 2048-ANDROID APPLICATIONS

	2048-android v1.0	2048-android v2.06
Functional Suitability	2,14	3,57
	Poor	Fair
Reliability	2,5	3,75
	Poor	Fair
Performance Efficiency	3,125	5,625
	Fair	Good
Portability	0	5
	Very Poor	Good
Maintainability	5,7	1,11
	Good	Poor
Data Integrity	0	3,63
	Very Poor	Fair
Usability	1	4
	Poor	Fair
Security	2,72	7,27
	Poor	Very Good
Compatibility	0	4,28
	Very Poor	Fair

4.3 GATHERING EXPERIMENT DATA FOR CASE STUDY

To gather open source applications, we used an application called F-Droid [51]. From F-Droid's list, we have searched for applications based on the following criteria:

- To be open source
- To be large (min 10000 KLOC)
- To have multiple releases.

We selected three Android applications for case study: Adblock Plus [55], KeePassDroid [56] and CosyDVR [57]. The first application, Adblock Plus consists of 11.055 KLOC and 10 releases. Adblock Plus is a free open source application that allows the user to block annoying advertisements, disable tracking and block domains, which is known to spread malware. The other application, KeePassDroid consists of 29.328 KLOC and 115 releases. KeePassDroid is a free open source password manager. Users can store all their passwords in one database, which is locked with one master key. Thus users have to remember only one password to reach the whole database. The third application, CosyDVR consists of 24.087 KLOC and 21 releases. CosyDVR is a free and open source DVR software kit designed for in-car use.

We downloaded source codes of the selected applications for the case study. We downloaded all releases of Adblock Plus, 10 out of the 115 releases of KeePassDroid, and 10 out of the 21 releases of CosyDVR. While selecting the 10

releases of KeePassDroid and CosyDVR, we considered the releases that have more commits. When some changes are made in application, commits are used to save the changes to application's repository. If the release has more commits, it means that it has more changes. We preferred to download 10 releases of each application to evaluate equal numbers of releases of the applications.

4.4 GATHERING AND NORMALIZING METRIC DATA

We calculated the 18 metrics that we described in Table 3.2 for the 10 releases of Adblock Plus, KeePassDroid and CosyDVR. In this study, for calculating Object-Oriented metrics we used Eclipse Metrics plug-in 1.3.8 [58]. For calculating the additional metrics that we identified, we implemented a Java program since there is no tool or plug-in developed specifically to calculate these metrics.

Using ECLIPSE plug-in and our Java program, we obtained the metric values. We provided them in Table 4.6, Table 4.7 and Table 4.8. Table 4.6 contains the metric values for Adblock Plus application. Table 4.7 contains the metric values for KeePassDroid application. Table 4.8 contains the metric values for CosyDVR application. If there was no metric value in the release of the application, we filled the related cells with the hyphen mark (-).

Table 4. 6: METRIC VALUES FOR ADBLOCK PLUS

	Adblock Plus v1.0	Adblock Plus v1.0.1	Adblock Plus v1.1	Adblock Plus v1.1.1	Adblock Plus v1.1.2	Adblock Plus v1.1.3	Adblock Plus v1.1.4	Adblock Plus v1.2	Adblock Plus v1.2.1	Adblock Plus v1.3
# Of Classes (NOC)	92	92	92	90	91	91	91	88	88	147
Depth of Inheritance Tree (DIT)	8	8	4	5	5	5	5	5	5	5
Instability (COP)	0.331	0.331	0.593	0.345	0.345	0.351	0.345	0.334	0.334	0.357
1 / Lack of Cohesion of Methods (COH)	5.714	5.714	5.154	5.814	5.882	5.882	5.848	5.747	5.747	4.807
# Of Attributes (NOAT)	218	218	218	218	218	218	218	259	259	346
# Of Methods (NOM)	418	418	418	418	419	419	420	422	422	644
McCabe Cyclomatic Complexity (CYC)	3.294	3.293	3.323	3.310	3.293	3.293	3.290	3.231	3.231	2.549
# Of Activities (NOAC)	3	3	4	4	4	4	4	4	4	4
# Of Services (NOS)	2	2	2	2	2	2	2	2	2	2
# Of Broadcast Receivers (NOBR)	2	2	2	2	2	2	2	2	2	3
# Of Content Providers (NOCP)	0	0	0	0	0	0	0	0	0	0
# Of Dialogs (NOD)	1	1	1	1	1	1	1	1	1	1
# Of Threads (NOT)	5	5	5	5	5	5	5	5	5	4
android:minSdkVersion (MSDK)	7	7	7	7	7	7	7	7	7	9
android:targetSdkVersion (TSDK)	7	7	16	16	16	16	16	16	16	16

# Of Intents (NOI)	23	25	29	31	31	31	32	34	34	37
# Of Tables in Database (NOTD)	0	0	0	0	0	0	0	0	0	0
# Of <uses-permission> (NOP)	3	3	4	4	4	4	4	4	4	5

Table 4. 7: METRIC VALUES FOR KEEPASSDROID

	KeePassDroid v0.0.1	KeePassDroid v0.2.0	KeePassDroid v0.6	KeePassDroid v1.0	KeePassDroid v1.6	KeePassDroid v1.9	KeePassDroid v1.99.0	KeePassDroid v1.99.9	KeePassDroid v2.0	KeePassDroid v2.0.4
# Of Classes (NOC)	60	92	125	109	110	359	425	429	444	450
Depth of Inheritance Tree (DIT)	6	7	8	9	10	10	10	10	10	10
Instability (COP)	0.603	0.470	0.433	0.285	0.402	0.416	0.428	0.434	0.436	0.430
1 / Lack of Cohesion of Methods (COH)	5.102	5.464	6.849	6.535	6.172	6.410	6.369	6.410	6.452	6.494
# Of Attributes (NOAT)	143	201	250	190	640	650	725	731	773	777
# Of Methods (NOM)	295	412	525	360	1664	1726	2098	2163	2212	2241
McCabe Cyclomatic Complexity (CYC)	1.84	1.866	1.836	1.799	2.03	2.04	2.01	2.02	2.04	2.05
# Of Activities (NOAC)	2	5	6	6	7	10	12	12	12	13
# Of Services (NOS)	0	1	1	1	1	1	1	1	1	1
# Of Broadcast Receivers (NOBR)	0	0	0	0	0	0	0	0	0	0
# Of Content Providers (NOCP)	0	0	0	0	0	0	0	0	0	0
# Of Dialogs (NOD)	0	1	4	5	6	5	5	5	5	5
# Of Threads (NOT)	0	0	0	0	0	0	0	0	0	0
android:minSdkVersion (MSDK)	1	1	3	3	3	3	3	3	3	3
android:targetSdkVersion (TSDK)	-	-	-	4	8	8	12	12	12	12
# Of Intents (NOI)	2	19	19	16	19	26	24	30	29	30
# Of Tables in Database (NOTD)	7	10	10	13	23	26	33	34	39	39
# Of <uses-permission> (NOP)	0	0	0	1	1	1	2	2	2	2

Table 4. 8: METRIC VALUES FOR COSYDVR

	CosyDVR v1.0	CosyDVR v1.0.3	CosyDVR v1.0.6	CosyDVR v1.1	CosyDVR v1.2	CosyDVR v1.3	CosyDVR v1.3.1	CosyDVR v1.3.5	CosyDVR v1.3.9	CosyDVR v1.3.11
# Of Classes (NOC)	23	23	25	25	25	25	25	25	26	60
Depth of Inheritance Tree (DIT)	7	7	7	7	7	7	7	7	7	6
Instability (COP)	1	1	1	1	1	1	1	1	1	0,603
1 / Lack of Cohesion of Methods (COH)	12,19	12,19	13,33	13,33	13,33	13,33	13,33	13,33	13,88	5,102
# Of Attributes (NOAT)	63	65	68	68	69	71	71	73	80	143
# Of Methods (NOM)	49	49	55	55	56	57	57	57	56	295
McCabe Cyclomatic Complexity (CYC)	2,184	2,204	2,364	2,364	2,429	2,421	2,421	2,526	2,789	1,84
# Of Activities (NOAC)	1	1	1	1	1	1	1	1	1	1
# Of Services (NOS)	1	1	1	1	1	1	1	1	1	1
# Of Broadcast Receivers (NOBR)	0	0	0	0	0	0	0	0	0	0
# Of Content Providers (NOCP)	0	0	0	0	0	0	0	0	0	0
# Of Dialogs (NOD)	0	0	0	0	0	0	0	0	0	0
# Of Threads (NOT)	0	0	0	0	0	0	0	0	0	0
android:minSdkVersion (MSDK)	11	11	11	11	11	16	16	16	16	16
android:targetSdkVersion (TSDK)	17	17	17	17	17	17	17	17	17	17
# Of Intents (NOI)	4	4	5	5	5	5	5	5	5	6
# Of Tables in Database (NOTD)	0	0	0	0	0	0	0	0	0	0
# Of <uses-permission> (NOP)	8	8	12	8	9	9	9	9	9	9

Since we combined actual metric values of different ranges in the computation of the quality characteristics, normalization of these metric values was necessary. Therefore, we normalized the metric values. We used Min-Max Normalization that is the process converting data to a value between 0 and 1.

We calculated the normalized value of the metric value X_i in the i_{th} row as [59]:

$$X_{i,\ 0\ \text{to}\ 1} = \frac{X_i - X_{Min}}{X_{Max} - X_{Min}}$$

where

X_{min} = the minimum value for variable X

X_{max} = the maximum value for variable X

We did the normalization of the values of three applications separately because three independent vendors have developed the applications. Table 4.9, Table 4.10 and Table 4.11 show the normalized metric values for Android Applications.

Table 4. 9: NORMALIZED METRIC VALUES FOR ADBLOCK PLUS

	Adblock Plus v1.0	Adblock Plus v1.0.1	Adblock Plus v1.1	Adblock Plus v1.1.1	Adblock Plus v1.1.2	Adblock Plus v1.1.3	Adblock Plus v1.1.4	Adblock Plus v1.2	Adblock Plus v1.2.1	Adblock Plus v1.3
# Of Classes (NOC)	0,067	0,067	0,067	0,033	0,050	0,050	0,050	0	0	1
Depth of Inheritance Tree (DIT)	1	1	0	0,25	0,25	0,25	0,25	0,25	0,25	0,25
Instability (COP)	0	0	1	0,053	0,053	0,076	0,053	0,011	0,011	0,099
1 / Lack of Cohesion of Methods (COH)	0,843	0,843	0,322	0,936	1	1	0,968	0,874	0,874	0
# Of Attributes (NOAT)	0	0	0	0	0	0	0	0,320	0,320	1
# Of Methods (NOM)	0	0	0	0	0,004	0,004	0,008	0,017	0,017	1
McCabe Cyclomatic Complexity (CYC)	0,962	0,961	1	0,983	0,961	0,961	0,957	0,881	0,881	0
# Of Activities (NOAC)	0	0	1	1	1	1	1	1	1	1
# Of Services (NOS)	0	0	0	0	0	0	0	0	0	0
# Of Broadcast Receivers (NOBR)	0	0	0	0	0	0	0	0	0	1
# Of Content Providers (NOCP)	0	0	0	0	0	0	0	0	0	0
# Of Dialogs (NOD)	0	0	0	0	0	0	0	0	0	0
# Of Threads (NOT)	1	1	1	1	1	1	1	1	1	0
android:minSdkVersion (MSDK)	0	0	0	0	0	0	0	0	0	1
android:targetSdkVersion (TSDK)	0	0	1	1	1	1	1	1	1	1
# Of Intents (NOI)	0	0,142	0,428	0,571	0,571	0,571	0,642	0,785	0,785	1
# Of Tables in Database (NOTD)	0	0	0	0	0	0	0	0	0	0
# Of <uses-permission> (NOP)	0	0	0,5	0,5	0,5	0,5	0,5	0,5	0,5	1

Table 4. 10: NORMALIZED METRIC VALUES FOR KEEPASSDROID

	KeePassDroid v0.0.1	KeePassDroid v0.2.0	KeePassDroid v0.6	KeePassDroid v1.0	KeePassDroid v1.6	KeePassDroid v1.9	KeePassDroid v1.99.0	KeePassDroid v1.99.9	KeePassDroid v2.0	KeePassDroid v2.0.4
# Of Classes (NOC)	0	0,082	0,166	0,125	0,128	0,766	0,935	0,946	0,984	1
Depth of Inheritance Tree (DIT)	0	0,25	0,5	0,75	1	1	1	1	1	1
Instability (COP)	1	0,581	0,465	0	0,367	0,411	0,449	0,468	0,474	0,455
1 / Lack of Cohesion of Methods (COH)	0	0,207	1	0,82	0,612	0,748	0,725	0,748	0,772	0,796
# Of Attributes (NOAT)	0	0,091	0,168	0,074	0,783	0,799	0,917	0,927	0,993	1
# Of Methods (NOM)	0	0,06	0,118	0,033	0,703	0,735	0,926	0,959	0,985	1
McCabe Cyclomatic Complexity (CYC)	0,163	0,266	0,147	0	0,92	0,96	0,84	0,88	0,96	1
# Of Activities (NOAC)	0	0,272	0,363	0,363	0,454	0,727	0,909	0,909	0,909	1
# Of Services (NOS)	0	1	1	1	1	1	1	1	1	1
# Of Broadcast Receivers (NOBR)	0	0	0	0	0	0	0	0	0	0
# Of Content Providers (NOCP)	0	0	0	0	0	0	0	0	0	0
# Of Dialogs (NOD)	0	0,2	0,8	1	1,2	1	1	1	1	1
# Of Threads (NOT)	0	0	0	0	0	0	0	0	0	0
android:minSdkVersion (MSDK)	0	0	1	1	1	1	1	1	1	1
android:targetSdkVersion (TSDK)	0	0	0	0,333	0,666	0,666	1	1	1	1
# Of Intents (NOI)	0	0,607	0,607	0,5	0,607	0,857	0,785	1	0,964	1
# Of Tables in Database (NOTD)	0	0,093	0,093	0,187	0,5	0,593	0,812	0,843	1	1
# Of <uses-permission> (NOP)	0	0	0	0,5	0,5	0,5	1	1	1	1

Table 4. 11: NORMALIZED METRIC VALUES FOR COSYDVR

	CosyDVR v1.0	CosyDVR v1.0.3	CosyDVR v1.0.6	CosyDVR v1.1	CosyDVR v1.2	CosyDVR v1.3	CosyDVR v1.3.1	CosyDVR v1.3.5	CosyDVR v1.3.9	CosyDVR v1.3.11
# Of Classes (NOC)	0	0	0,054	0,054	0,054	0,054	0,054	0,054	0,081	1
Depth of Inheritance Tree (DIT)	1	1	1	1	1	1	1	1	1	0
Instability (COP)	1	1	1	1	1	1	1	1	1	0
1 / Lack of Cohesion of Methods (COH)	0,807	0,807	0,937	0,937	0,937	0,937	0,937	0,937	1	0
# Of Attributes (NOAT)	0	0,025	0,062	0,062	0,075	0,1	0,1	0,125	0,212	1
# Of Methods (NOM)	0	0	0,024	0,024	0,028	0,032	0,032	0,032	0,028	1
McCabe Cyclomatic Complexity (CYC)	0,362	0,383	0,552	0,552	0,620	0,612	0,612	0,722	1	0
# Of Activities (NOAC)	0	0	0	0	0	0	0	0	0	0
# Of Services (NOS)	0	0	0	0	0	0	0	0	0	0
# Of Broadcast Receivers (NOBR)	0	0	0	0	0	0	0	0	0	0
# Of Content Providers (NOCP)	0	0	0	0	0	0	0	0	0	0
# Of Dialogs (NOD)	0	0	0	0	0	0	0	0	0	0
# Of Threads (NOT)	0	0	0	0	0	0	0	0	0	0
android:minSdkVersion (MSDK)	0	0	0	0	0	1	1	1	1	1
android:targetSdkVersion (TSDK)	0	0	0	0	0	0	0	0	0	0
# Of Intents (NOI)	0	0	0,5	0,5	0,5	0,5	0,5	0,5	0,5	1
# Of Tables in Database (NOTD)	0	0	0	0	0	0	0	0	0	0
# Of <uses-permission> (NOP)	0	0	1	0	0,25	0,25	0,25	0,25	0,25	0,25

4.5 ANALYZING RESULTS

We calculated the values of identified nine quality characteristics by using the computation formulas stated in Table 3.4. We used normalized values of the metrics for each of the formula. Table 4.12, Table 4.13 and Table 4.14 shows the computed values of nine quality characteristics for the three Android applications based on the normalized metric values.

Table 4. 12: COMPUTED QUALITY CHARACTERISTICS FOR ADBLOCK PLUS

	Adblock Plus v1.0	Adblock Plus v1.0.1	Adblock Plus v1.1	Adblock Plus v1.1.1	Adblock Plus v1.1.2	Adblock Plus v1.1.3	Adblock Plus v1.1.4	Adblock Plus v1.2	Adblock Plus v1.2.1	Adblock Plus v1.3
Functional Suitability	0,291	0,305	0,281	0,473	0,482	0,479	0,486	0,491	0,491	0,715
Reliability	0,210	0,210	0,080	0,408	0,424	0,418	0,416	0,483	0,483	0,412
Performance Efficiency	-0,447	-0,447	0,158	0,025	0,044	0,044	0,046	-0,020	-0,020	0,375
Portability	0,187	0,210	0,092	0,146	0,152	0,152	0,165	0,173	0,173	0,371
Maintainability	0,289	0,289	0,149	0,312	0,315	0,311	0,315	0,315	0,315	0,441
Data Integrity	0,255	0,268	0,427	0,432	0,439	0,441	0,442	0,473	0,473	0,481
Usability	0,133	0,151	0,249	0,263	0,265	0,265	0,275	0,327	0,327	0,5
Security	0,182	0,210	0,463	0,418	0,435	0,440	0,444	0,501	0,501	1,019
Compatibility	0,258	0,297	0,304	0,333	0,342	0,345	0,357	0,423	0,423	0,468

Table 4. 13: COMPUTED QUALITY CHARACTERISTICS FOR KEEPASSDROID

	KeePassDroid v0.0.1	KeePassDroid v0.2.0	KeePassDroid v0.6	KeePassDroid v1.0	KeePassDroid v1.6	KeePassDroid v1.9	KeePassDroid v1.99.0	KeePassDroid v1.99.9	KeePassDroid v2.0	KeePassDroid v2.0.3
Functional Suitability	-0,1	0,199	0,438	0,511	0,630	0,768	0,864	0,893	0,914	0,934
Reliability	-0,25	0,116	0,300	0,286	0,257	0,284	0,298	0,301	0,322	0,335
Performance Efficiency	-0,081	-0,636	-0,979	-1,049	-2,066	-1,718	-1,587	-1,620	-1,736	-1,75
Portability	0	0,358	0,107	0,118	0,340	0,613	0,724	0,774	0,811	0,825
Maintainability	-0,2	0,111	0,107	0,178	0,002	0,188	0,256	0,254	0,286	0,309
Data Integrity	0,09	0,308	0,420	0,368	0,583	0,683	0,758	0,788	0,810	0,832
Usability	0	0,289	0,402	0,324	0,546	0,673	0,684	0,717	0,729	0,75
Security	0,2	0,088	0,267	0,135	0,34	0,526	0,622	0,672	0,634	0,650
Compatibility	0,14	0,341	0,481	0,396	0,626	0,737	0,766	0,838	0,863	0,875

Table 4. 14: COMPUTED QUALITY CHARACTERISTICS FOR COSYDVR

	CosyDVR v1.0	CosyDVR v1.0.3	CosyDVR v1.0.6	CosyDVR v1.1	CosyDVR v1.2	CosyDVR v1.3	CosyDVR v1.3.1	CosyDVR v1.3.5	CosyDVR v1.3.9	CosyDVR v1.3.11
Functional Suitability	0,080	0,080	0,151	0,151	0,151	0,252	0,252	0,252	0,260	0,4
Reliability	-0,298	-0,292	-0,250	-0,250	-0,247	-0,240	-0,240	-0,234	-0,197	0,25
Performance Efficiency	-0,681	-0,697	-0,514	-0,764	-0,739	-0,741	-0,741	-0,802	-0,95	0,312
Portability	0,165	0,165	0,269	0,269	0,269	-0,059	-0,059	-0,059	-0,051	0,33
Maintainability	-0,4	-0,4	-0,394	-0,394	-0,394	-0,395	-0,395	-0,395	-0,389	0
Data Integrity	0,252	0,254	0,317	0,317	0,318	0,321	0,321	0,323	0,336	0,27
Usability	0	0,003	-0,045	0,08	0,05	0,054	0,054	0,057	0,071	0,468
Security	0,361	0,366	0,715	0,515	0,568	0,574	0,574	0,579	0,614	0,85
Compatibility	0,392	0,396	0,559	0,559	0,561	0,565	0,565	0,568	0,589	0,42

In order to identify the categories of quality characteristics of the releases, we changed the ranges of computed quality characteristics to [0, 10]. We stated the formula used for changing the ranges of quality characteristics in Section 4.1.

When all quality characteristics were converted to the range of [0, 10], we applied our rating scale to these values. The rating scale contains six categories: 0-1 Very Poor, 1-3 Poor, 3-5 Fair, 5-7 Good, 7-9 Very Good and 9-10 Excellent. We illustrated the rating scale in Section 4.1.1.

Table 4.15, Table 4.16 and Table 4.17 shows the changed values of nine quality characteristics for the three Android applications based on the ranges of [0,10] and the categories they entered according to the determined rating scale.

Table 4. 15: COMPUTED QUALITY CHARACTERISTICS FOR ADBLOCK PLUS WITHIN THE RANGE [0, 10] AND THE CATEGORIES OF THE VALUES

	Adblock Plus v1.0	Adblock Plus v1.0.1	Adblock Plus v1.1	Adblock Plus v1.1.1	Adblock Plus v1.1.2	Adblock Plus v1.1.3	Adblock Plus v1.1.4	Adblock Plus v1.2	Adblock Plus v1.2.1	Adblock Plus v1.3
Functional Suitability	3,507	3,607	3,435	4,807	4,871	4,85	4,9	4,935	4,935	6,535
	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Good
Reliability	3,55	3,55	2,9	4,54	4,62	4,59	4,58	4,915	4,915	4,56
	Fair	Fair	Poor	Fair	Fair	Fair	Fair	Fair	Fair	Fair
Performance Efficiency	5,132	5,132	6,645	6,312	6,36	6,36	6,365	6,2	6,2	7,187
	Good	Good	Good	Good	Good	Good	Good	Good	Good	Very Good
Portability	3,133	3,272	2,557	2,884	2,921	2,921	3	3,048	3,048	4,248
	Fair	Fair	Poor	Poor	Poor	Poor	Fair	Fair	Fair	Fair
Maintainability	3,827	3,827	3,05	3,955	3,972	3,95	3,972	3,972	3,972	4,672
	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair
Data Integrity	2,55	2,68	4,27	4,32	4,39	4,41	4,42	4,73	4,73	4,81
	Poor	Poor	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair
Usability	2,064	2,208	2,992	3,104	3,12	3,12	3,2	3,616	3,616	5
	Poor	Poor	Poor	Fair	Fair	Fair	Fair	Fair	Fair	Good
Security	3,554	3,681	4,831	4,627	4,704	4,727	4,745	5,004	5,004	7,359
	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Good	Good	Very Good
Compatibility	2,58	2,97	3,04	3,33	3,42	3,45	3,57	4,23	4,23	4,68
	Poor	Poor	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair

Table 4. 16: COMPUTED QUALITY CHARACTERISTICS FOR KEEPASSDROID WITHIN THE RANGE [0, 10] AND THE CATEGORIES OF THE VALUES

	KeePassDroid v0.0.1	KeePassDroid v0.2.0	KeePassDroid v0.6	KeePassDroid v1.0	KeePassDroid v1.6	KeePassDroid v1.9	KeePassDroid v1.99.0	KeePassDroid v1.99.9	KeePassDroid v2.0	KeePassDroid v2.0.3
Functional Suitability	0,714	2,85	4,557	5,078	5,928	6,914	7,6	7,807	7,957	8,1
	Very Poor	Poor	Fair	Good	Good	Good	Very Good	Very Good	Very Good	Very Good
Reliability	1,25	3,08	4	3,93	3,785	3,92	3,99	4,005	4,11	4,175
	Poor	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair
Performance Efficiency	6,047	4,66	3,802	3,627	1,085	1,955	2,282	2,2	1,91	1,875
	Good	Fair	Fair	Fair	Poor	Poor	Poor	Poor	Poor	Poor
Portability	2	4,169	2,648	2,715	4,060	5,715	6,387	6,690	6,915	7
	Poor	Fair	Poor	Poor	Fair	Good	Good	Good	Good	Very Good
Maintainability	1,111	2,838	2,816	3,211	2,233	3,266	3,644	3,633	3,811	3,938
	Poor	Poor	Poor	Fair	Poor	Fair	Fair	Fair	Fair	Fair
Data Integrity	0,9	3,08	4,2	3,68	5,83	6,83	7,58	7,88	8,1	8,32
	Very Poor	Fair	Good	Fair	Good	Good	Very Good	Very Good	Very Good	Very Good
Usability	1	3,312	4,216	3,592	5,368	6,384	6,472	6,736	6,832	7
	Poor	Fair	Fair	Fair	Good	Good	Good	Good	Good	Very Good
Security	3,636	3,127	3,940	3,340	4,272	5,118	5,554	5,781	5,609	5,681
	Fair	Fair	Fair	Fair	Fair	Good	Good	Good	Good	Good
Compatibility	1,4	3,41	4,81	3,96	6,26	7,37	7,66	8,38	8,63	8,75
	Poor	Fair	Fair	Fair	Good	Very Good	Very Good	Very Good	Very Good	Very Good

Table 4. 17: COMPUTED QUALITY CHARACTERISTICS FOR COSYDVR WITHIN THE RANGE [0, 10] AND THE CATEGORIES OF THE VALUES

	CosyDVR v1.0	CosyDVR v1.0.3	CosyDVR v1.0.6	CosyDVR v1.1	CosyDVR v1.2	CosyDVR v1.3	CosyDVR v1.3.1	CosyDVR v1.3.5	CosyDVR v1.3.9	CosyDVR v1.3.11
Functional Suitability	2	2	2,507	2,507	2,507	3,228	3,228	3,228	3,285	4,285
	Poor	Poor	Poor	Poor	Poor	Fair	Fair	Fair	Fair	Fair
Reliability	1,01	1,04	1,25	1,25	1,265	1,3	1,3	1,33	1,515	3,75
	Poor	Poor	Poor	Poor	Poor	Poor	Poor	Poor	Poor	Fair
Performance Efficiency	4,547	4,507	4,965	4,34	4,402	4,397	4,397	4,245	3,875	7,03
	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Very Good
Portability	3	3	3,630	3,630	3,630	1,642	1,642	1,642	1,690	4
	Poor	Poor	Fair	Fair	Fair	Poor	Poor	Poor	Poor	Fair
Maintainability	0	0	0,033	0,033	0,033	0,027	0,027	0,027	0,061	2,222
	Very Poor	Very Poor	Very Poor	Very Poor	Very Poor	Very poor	Very Poor	Very Poor	Very Poor	Poor
Data Integrity	2,52	2,54	3,17	3,17	3,18	3,21	3,21	3,23	3,36	2,7
	Poor	Poor	Fair	Fair	Fair	Fair	Fair	Fair	Fair	Poor
Usability	1	1,024	0,64	1,64	1,4	1,432	1,432	1,456	1,568	4,744
	Very Poor	Poor	Very Poor	Poor	Poor	Poor	Poor	Poor	Poor	Fair
Security	4,368	4,390	5,977	5,068	5,309	5,336	5,336	5,359	5,518	6,590
	Fair	Fair	Good	Good	Good	Good	Good	Good	Good	Good
Compatibility	3,92	3,96	5,59	5,59	5,61	5,65	5,65	5,68	5,89	4,2
	Fair	Fair	Good	Good	Good	Good	Good	Good	Good	Fair

We drew the plots of quality characteristics based on these computed values. We provided the plots of quality characteristics of Adblock Plus, KeePassDroid and CosyDVR applications in Figure 4.3, Figure 4.4 and Figure 4.5.

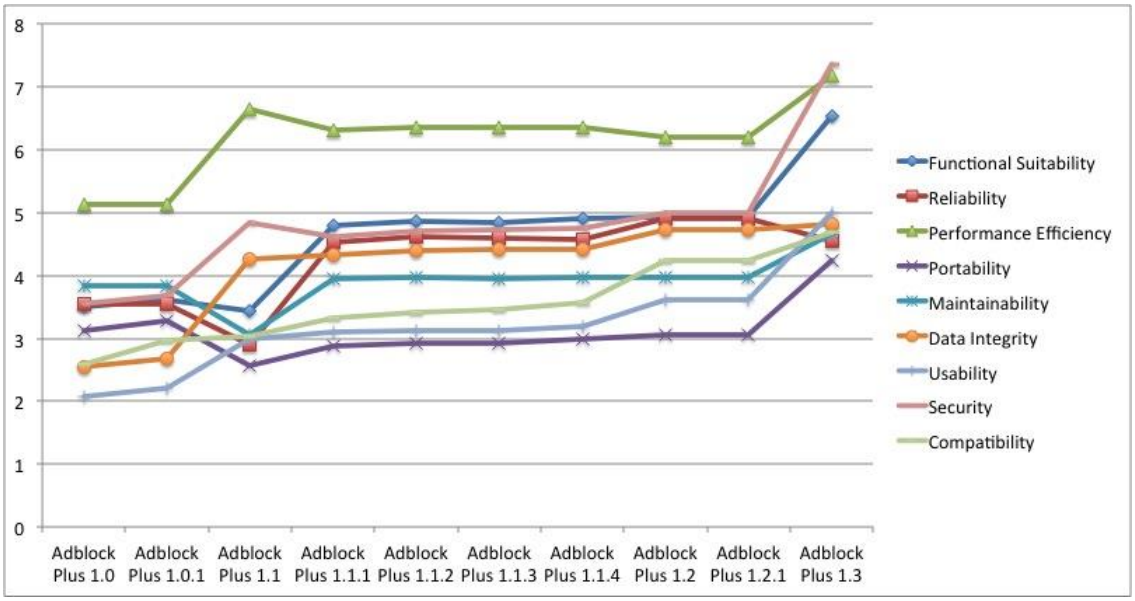


Figure 4. 3: Plots Of Quality Characteristics Of Adblock Plus

Quality scores of all quality characteristics, which are Functional Suitability, Reliability, Performance Efficiency, Portability, Maintainability, Data Integrity, Usability, Security, and Compatibility, of Adblock Plus application increased.

- **Quality Score of Functional Suitability:** According to the Functional Suitability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Degree of Cohesion (COH), Number of Methods (NOM), Number of Activities (NOAC), Number of Services (NOS), Number of Broadcast Receivers (NOBR), Number of Threads (NOT), Android Min SDK Version (MSDK), Android Target SDK Version (TSDK), Number of Intents (NOI) and Number of Tables in Database (NOTD) metrics affects characteristic positively; Number of Content Providers (NOCP) and Degree of Coupling (COP) metrics affect the characteristic negatively. The metric values of NOC, COP, NOM, NOAC, NOBR, MSDK, TSDK, and NOI increased; DIT, COH, and NOT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Reliability:** According to the Reliability formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT) and Degree of Coupling (COP) metrics affects Functional Suitability characteristic negatively; Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Services (NOS), Number of Broadcast Receivers (NOBR), Number of Content Providers (NOCP) and Number of Threads (NOT) metrics affect the characteristic positively. The metric values of COP, NOAT, and NOBR increased; DIT, COH, and NOT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

    1. The reason for the sudden decrease in the quality scores of Reliability characteristic between Adblock Plus v1.0.1 and v1.1 is the decrease in the degree of coupling, which was identified as Instability (COP) metric in our study. COP value is 0.331 in Adblock Plus v1.0.1 and 0.593 in Adblock Plus v1.1.

       The main reason of the decrease in Instability (COP) metric is most probably the decrease in Depth of Inheritance Tree (DIT) metric. Developers of Adblock Plus probably decreased the maximum length of a path from a class to the root class in the inheritance hierarchy to reduce the complexity of the application. When DIT metric was 8 in Adblock Plus v1.0.1, it was more difficult to make changes without impacting the rest of the application. Application was more stable. DIT metric was 4 in Adblock Plus v1.1; means making changes on application became easier. Instability of application increased.

       As can be understood from the formula of Reliability, COP metric affects quality score of the characteristic negatively. The increase in this metric value reduced the quality score of Reliability characteristic.

2. The reason for the decrease in the quality scores of Reliability between Adblock Plus v1.2.1 and v1.3 is the decrease in the value of Cohesion (COH) and increase in the value of Instability (COP). COH metric affects the quality score positively and COP metric affects negatively. The changes in the values of metrics caused the decrease in total.

   It is impossible to arrive any definite outcome about the main reasons of the decrease in the value of COH metric and the increase in the value of COP metric. It is probably because of the decrease in linearly independent paths through source code. McCabe Cyclomatic Complexity (CYC) measures the linearly independent paths. The decrease in CYC metric indicates that the dependency increased. The increase of dependency may be the reason of the decrease in cohesion. Apart from that, the increase in class number and method number may also the reason for the decrease of cohesion. Because cohesion measures how well the methods of a class are related to each other. Low cohesion often correlates with high coupling [46]. The main reason of the increase in COP metric may be the decrease in COH metric.

- **Quality Score of Performance Efficiency:** According to the Performance Efficiency formula in Table 3.4, the increase in Number of Classes (NOC), Number of Content Providers (NOCP), Number of Threads (NOT) and Number of Permissions (NOP) metrics affect the characteristic positively; Depth of Inheritance Tree (DIT), Number of Attributes (NOAT), McCabe Cyclomatic Complexity (CYC), Number of Services (NOS), Number of Dialogs (NOD), Number of Tables in Database (NOTD) affect the characteristic negatively. The metric values of NOC, NOAT, and NOP increased; DIT, CYC, and NOT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Portability:** According to the Portability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Number of Methods (NOM), Number of Services (NOS), Number of Content Providers (NOCP), Number of Intents (NOI), Number of Tables in Database (NOTD) metrics affect the characteristic positively; Android Min SDK Version (MSDK) metric affects the characteristic negatively. The metric values of NOC, NOM, MSDK, and NOI increased; DIT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

  1. The reason for the sudden decrease in the quality scores of Portability between Adblock Plus v1.0.1 and v1.1 is the decrease in the value of Depth of Inheritance Tree (DIT) metric. DIT value is 8 in Adblock Plus v1.0.1 and 4 in Adblock Plus v1.1. As DIT metric affects quality

score of Portability positively, the decrease in this metric value reduced the quality score of Portability characteristic.

The main reason of the decrease in the value of DIT metric is probably the preference of the developers. Developers of Adblock Plus probably decreased the maximum length of a path from a class to the root class in the inheritance hierarchy to reduce the complexity of the application.

- **Quality Score of Maintainability:** According to the Maintainability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Number of Activities (NOAC), Number of Services (NOS), Number of Content Providers (NOCP), Number of Threads (NOT) and Number of Tables in Database (NOTD) metrics affect the characteristic positively; Degree of Coupling (COP) and Number of Methods (NOM) metrics affect the characteristic negatively. The metric values of NOC, COP, NOM, and NOAC increased; DIT and NOT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

    1. The reason for the sudden decrease in the quality scores of Maintainability characteristic between Adblock Plus v1.0.1 and v1.1 is the decrease in the degree of coupling, which was identified as Instability (COP) metric in our study. COP value is 0.331 in Adblock Plus v1.0.1 and 0.593 in Adblock Plus v1.1.

       The main reason of the decrease in Instability (COP) metric is most probably the decrease in Depth of Inheritance Tree (DIT) metric. Developers of Adblock Plus probably decreased the maximum length of a path from a class to the root class in the inheritance hierarchy to reduce the complexity of the application. When DIT metric was 8 in Adblock Plus v1.0.1, it was more difficult to make changes without impacting the rest of the application. Application was more stable. DIT metric was 4 in Adblock Plus v1.1; means making changes on application became easier. Instability of application increased.

       As can be understood from the formula of Maintainability, COP metric affects quality score of the characteristics negatively. The increase in this metric value reduced the quality score of Maintainability characteristic.

- **Quality Score of Data Integrity:** According to the Data Integrity formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Methods (NOM), Number of Activities (NOAC), Number of Threads (NOT), Number of Intents (NOI) metrics affect the characteristic positively. The metric values of COP, NOAT, NOM, NOAC, and NOI increased; DIT, COH, and NOT decreased from the first release to the last

release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Usability:** According to the Usability formula in Table 3.4, the increase in Number of Classes (NOC), Number of Attributes (NOAT), Number of Methods (NOM), Number of Activities (NOAC), Number of Services (NOS), Number of Content Providers (NOCP), Number of Dialogs (NOD), Number of Threads (NOT), Number of Intents (NOI) metrics affect the characteristic positively; Number of Permissions (NOP) metric affects the characteristic negatively. The metric values of NOC, NOAT, NOM, NOAC, NOI, and NOP increased; NOT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Security:** According to the Security formula in Table 3.4, the increase in Number of Classes (NOC), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Content Providers (NOCP), Number of Intents (NOI) and Number of Permissions (NOP) metrics affect the characteristic positively; Number of Services (NOS), Number of Tables in Database (NOTD) metrics affects the characteristic negatively. The metric values of NOC, COP, NOAT, NOM, NOI, and NOP increased; COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Compatibility:** According to the Compatibility formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Intents (NOI), Number of Tables in Database (NOTD) metrics affects the characteristic positively. The metric values of COP, NOAT, NOI, and NOTD increased; DIT and COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

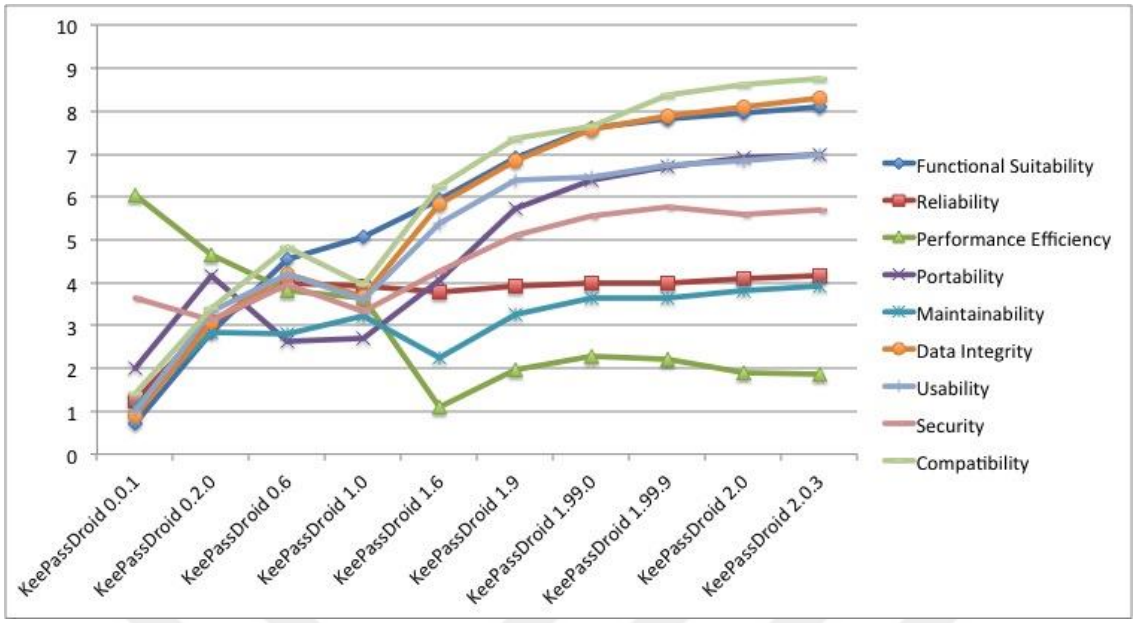


Figure 4. 4: Plots Of Quality Characteristics Of KeePassDroid

Quality scores of Functional Suitability, Reliability, Portability, Maintainability, Data Integrity, Usability, Security and Compatibility characteristics of KeePassDroid application increased; Performance Efficiency characteristic of KeePassDroid application decreased.

- **Quality score of Performance Efficiency:** According to the Performance Efficiency formula in Table 3.4, the increase in Number of Classes (NOC), Number of Content Providers (NOCP), Number of Threads (NOT) and Number of Permissions (NOP) metrics affect the characteristic positively; Depth of Inheritance Tree (DIT), Number of Attributes (NOAT), McCabe Cyclomatic Complexity (CYC), Number of Services (NOS), Number of Dialogs (NOD), Number of Tables in Database (NOTD) affect the characteristic negatively. The metric values of NOC, DIT, NOAT, CYC, NOS, NOD, NOTD, NOP increased from the first release to the last release of the application. The changes in the values of metrics caused the decrease in total.

    1. The reason for the sudden decrease between KeePassDroid v1.0 and v1.6 is the three times increase in Number of Attributes (NAO) metric; this degree of change did not occur in other metric values. NOA value is 190 in KeePassDroid v1.0 and 640 in KeePassDroid v1.6.

       The main reason of the increase in attribute number is most probably the increase in class number and method number. Developers of KeePassDroid added new classes and methods to the application to update the application. These new classes and methods caused the increase in the number of attributes.

As NOA metric affects quality score of Performance Efficiency characteristic negatively, the increase in this metric value reduced the quality score of Performance Efficiency characteristic.

- **Quality Score of Functional Suitability:** According to the Functional Suitability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Degree of Cohesion (COH), Number of Methods (NOM), Number of Activities (NOAC), Number of Services (NOS), Number of Broadcast Receivers (NOBR), Number of Threads (NOT), Android Min SDK Version (MSDK), Android Target SDK Version (TSDK), Number of Intents (NOI) and Number of Tables in Database (NOTD) metrics affects characteristic positively; Number of Content Providers (NOCP) and Degree of Coupling (COP) metrics affect the characteristic negatively. The metric values of NOC, DIT, COH, NOM, NOAC, NOS, MSDK, TSDK, NOI, and NOTD increased; COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Reliability:** According to the Reliability formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT) and Degree of Coupling (COP) metrics affects Functional Suitability characteristic negatively; Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Services (NOS), Number of Broadcast Receivers (NOBR), Number of Content Providers (NOCP) and Number of Threads (NOT) metrics affect the characteristic positively. The metric values of DIT, COH, NOAT, and NOS increased; COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Portability:** According to the Portability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Number of Methods (NOM), Number of Services (NOS), Number of Content Providers (NOCP), Number of Intents (NOI), Number of Tables in Database (NOTD) metrics affect the characteristic positively; Android Min SDK Version (MSDK) metric affects the characteristic negatively. The metric values of NOC, DIT, NOM, NOS, MSDK, NOI, and NOTD increased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

    1. The reason for the sudden decrease in the quality scores of Portability between KeePassDroid v0.2.0 and v0.6 is the increase in the value of Android Min SDK Version (MSDK) metric. MSDK value is;
       - 1 (supported by platform version 1.0) in KeePassDroid v0.2.0, which was developed on May 14, 2009.
       - 3 (supported by platform versions 1.5) in KeePassDroid v0.6, which was developed on Sep 25, 2009.
       The change in MSDK value is connected with the changes on Android releases, because Android 1.0 was released on Sep 23, 2008 and Android 1.5 (Cupcake) was released on Apr 30, 2009 [63]. When developers of KeePassDroid developed the 0.2.0 version of the

application, Android 1.0 platform dominated the Android market. Four months later, they developed KeePassDroid v0.6 and updated the MSDK value to be supported by Android 1.5 platform. The developers of CosyDVR application probably changed the MSDK value to update their application in order to keep up with the latest developments.

As MSDK metric affect quality score of Portability negatively, the increase in this metric value reduced the quality score of Portability characteristic.

- **Quality Score of Maintainability:** According to the Maintainability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Number of Activities (NOAC), Number of Services (NOS), Number of Content Providers (NOCP), Number of Threads (NOT) and Number of Tables in Database (NOTD) metrics affect the characteristic positively; Degree of Coupling (COP) and Number of Methods (NOM) metrics affect the characteristic negatively. The metric values of NOC, DIT, NOM, NOAC, NOS, and NOTD increased; COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Data Integrity:** According to the Data Integrity formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Methods (NOM), Number of Activities (NOAC), Number of Threads (NOT), Number of Intents (NOI) metrics affect the characteristic positively. The metric values of DIT, COH, NOAT, NOM, NOAC, NOS, NOI, and NOTD increased; COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Usability:** According to the Usability formula in Table 3.4, the increase in Number of Classes (NOC), Number of Attributes (NOAT), Number of Methods (NOM), Number of Activities (NOAC), Number of Services (NOS), Number of Content Providers (NOCP), Number of Dialogs (NOD), Number of Threads (NOT), Number of Intents (NOI) metrics affect the characteristic positively; Number of Permissions (NOP) metric affects the characteristic negatively. The metric values of NOC, NOAT, NOM, NOAC, NOS, NOD, NOI, and NOP increased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Security:** According to the Security formula in Table 3.4, the increase in Number of Classes (NOC), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Content Providers (NOCP), Number of Intents (NOI) and Number of Permissions (NOP) metrics affect the characteristic positively; Number of Services (NOS), Number of Tables in Database (NOTD) metrics affects the

characteristic negatively. The metric values of NOC, COH, NOAT, NOM, NOS, NOI, NOTD, and NOP increased; COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Compatibility:** According to the Compatibility formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Intents (NOI), Number of Tables in Database (NOTD) metrics affects the characteristic positively. The metric values of DIT, COH, NOAT, NOI, and NOTD increased; COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

    1. The reason for the sudden decrease in the quality scores of Compatibility between KeePassDroid v0.6 and v1.0 is the decrease in the values of COP, NOAT and NOI metrics. Because COP, NOAT and NOI metrics affect quality score of Compatibility positively, the decrease in these metric values reduced the quality score of Compatibility characteristic.

       The main reason of the decrease in COP metric is most probably the decrease in Depth of Inheritance Tree (DIT) metric. Developers of KeePassDroid probably decreased the maximum length of a path from a class to the root class in the inheritance hierarchy to reduce the complexity of the application. When DIT metric was 8 in KeePassDroid v0.6, it was easier to make changes without impacting the rest of the application. Application was more instable. DIT metric was 9 in KeePassDroid v1.0; means making changes on application became more difficult. Instability of application decreased. The main reason of the decrease in NOAT and NOI metrics is probably the preference of the developers. Developers of KeePassDroid probably decreased the attribute number and intent number to simplify the application.

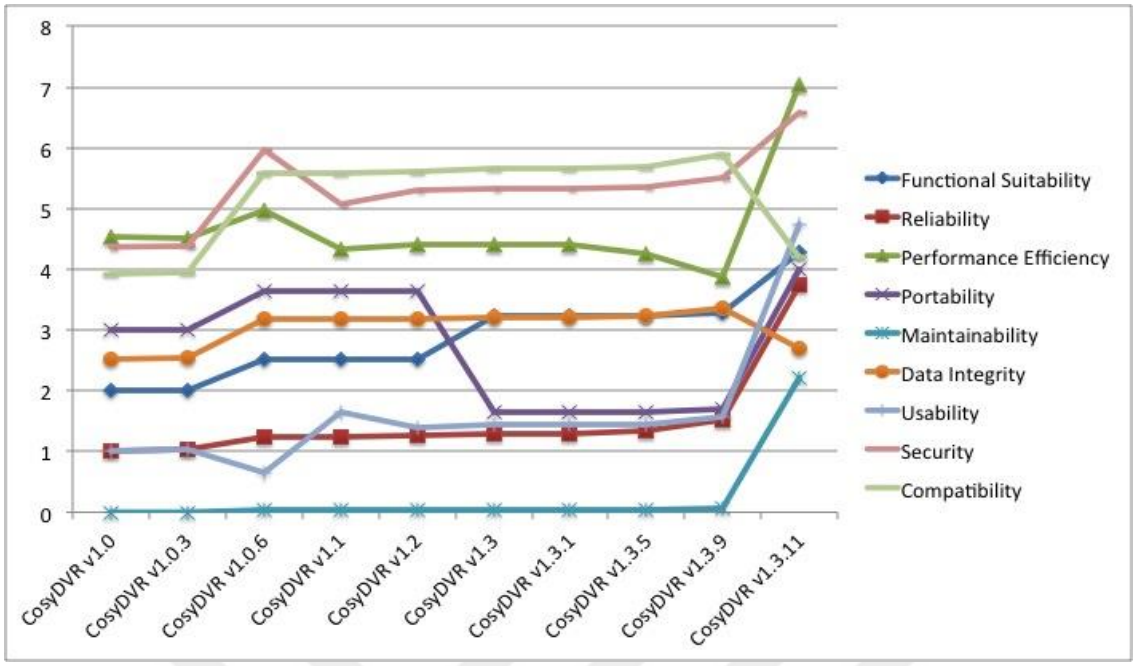


Figure 4. 5: Plots of Quality Characteristics of CosyDVR

- **Quality Score of Functional Suitability:** According to the Functional Suitability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Degree of Cohesion (COH), Number of Methods (NOM), Number of Activities (NOAC), Number of Services (NOS), Number of Broadcast Receivers (NOBR), Number of Threads (NOT), Android Min SDK Version (MSDK), Android Target SDK Version (TSDK), Number of Intents (NOI) and Number of Tables in Database (NOTD) metrics affects characteristic positively; Number of Content Providers (NOCP) and Degree of Coupling (COP) metrics affect the characteristic negatively. The metric values of NOC, NOM, MSDK and NOI increased; DIT, COP and COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Reliability:** According to the Reliability formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT) and Degree of Coupling (COP) metrics affects Functional Suitability characteristic negatively; Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Services (NOS), Number of Broadcast Receivers (NOBR), Number of Content Providers (NOCP) and Number of Threads (NOT) metrics affect the characteristic positively. The metric values of NOAT increased; DIT, COP, and COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Performance Efficiency:** According to the Performance Efficiency formula in Table 3.4, the increase in Number of Classes (NOC), Number of Content Providers (NOCP), Number of Threads (NOT) and Number of Permissions (NOP) metrics affect the characteristic positively; Depth of Inheritance Tree (DIT), Number of Attributes (NOAT), McCabe Cyclomatic Complexity (CYC), Number of Services (NOS), Number of

Dialogs (NOD), Number of Tables in Database (NOTD) affect the characteristic negatively. The metric values of NOC, NOAT, and NOP increased; DIT, and CYC decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Portability:** According to the Portability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Number of Methods (NOM), Number of Services (NOS), Number of Content Providers (NOCP), Number of Intents (NOI), Number of Tables in Database (NOTD) metrics affect the characteristic positively; Android Min SDK Version (MSDK) metric affects the characteristic negatively. The metric values of NOC, NOM, MSDK, and NOI increased; DIT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

    2. The reason for the sudden decrease in the quality scores of Portability between CosyDVR v1.2 and v1.3 is the increase in the value of Android Min SDK Version (MSDK) metric. MSDK value is;
    - 11 (supported by platform version 3.0.x) in CosyDVR v1.2, which was developed on 29 July 2014.
    - 16 (supported by platform versions 4.1 and 4.1.1) in CosyDVR v1.3, which was developed on 13 Nov 2014.
    The change in MSDK value is unconnected with the changes on Android releases, because Android 3.0 (Honeycomb) was released on 10 May 2011 and Android 4.1 (Jelly Bean) was released on 9 July 2012 [63]. The developers of CosyDVR application probably changed the MSDK value to update their application in order to keep up with the latest developments.

    As MSDK metric affects quality score of Portability negatively, the increase in this metric value reduced the quality score of Portability characteristic.

- **Quality Score of Maintainability:** According to the Maintainability formula in Table 3.4, the increase in Number of Classes (NOC), Depth of Inheritance Tree (DIT), Number of Activities (NOAC), Number of Services (NOS), Number of Content Providers (NOCP), Number of Threads (NOT) and Number of Tables in Database (NOTD) metrics affect the characteristic positively; Degree of Coupling (COP) and Number of Methods (NOM) metrics affect the characteristic negatively. The metric values of NOC, NOM, and NOAC increased; DIT and COP decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Data Integrity:** According to the Data Integrity formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Methods (NOM), Number of Activities (NOAC), Number of

Threads (NOT), Number of Intents (NOI) metrics affect the characteristic positively. The metric values of NOAT, NOM, NOAC, and NOI increased; DIT, COP, and COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Usability:** According to the Usability formula in Table 3.4, the increase in Number of Classes (NOC), Number of Attributes (NOAT), Number of Methods (NOM), Number of Activities (NOAC), Number of Services (NOS), Number of Content Providers (NOCP), Number of Dialogs (NOD), Number of Threads (NOT), Number of Intents (NOI) metrics affect the characteristic positively; Number of Permissions (NOP) metric affects the characteristic negatively. The metric values of NOC, NOAT, NOM, NOI, and NOP increased; NOT decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Security:** According to the Security formula in Table 3.4, the increase in Number of Classes (NOC), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Content Providers (NOCP), Number of Intents (NOI) and Number of Permissions (NOP) metrics affect the characteristic positively; Number of Services (NOS), Number of Tables in Database (NOTD) metrics affects the characteristic negatively. The metric values of NOC, NOAT, NOM, NOI, and NOP increased; COP and COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

- **Quality Score of Compatibility:** According to the Compatibility formula in Table 3.4, the increase in Depth of Inheritance Tree (DIT), Degree of Coupling (COP), Degree of Cohesion (COH), Number of Attributes (NOAT), Number of Intents (NOI), Number of Tables in Database (NOTD) metrics affects the characteristic positively. The metric values of NOAT, and NOI increased; DIT, COP and COH decreased from the first release to the last release of the application. The changes in the values of metrics caused the increase in total.

The expected increase in quality scores of quality characteristics verifies the prediction that these quality characteristics should increase in the new releases of Android applications. Most of the quality scores of the last releases of the Adblock Plus and KeePassDroid applications are in the categories more than or equal to "Fair" On the other hand quality scores of the last release of the CosyDVR are mostly less than or equal to "Fair".

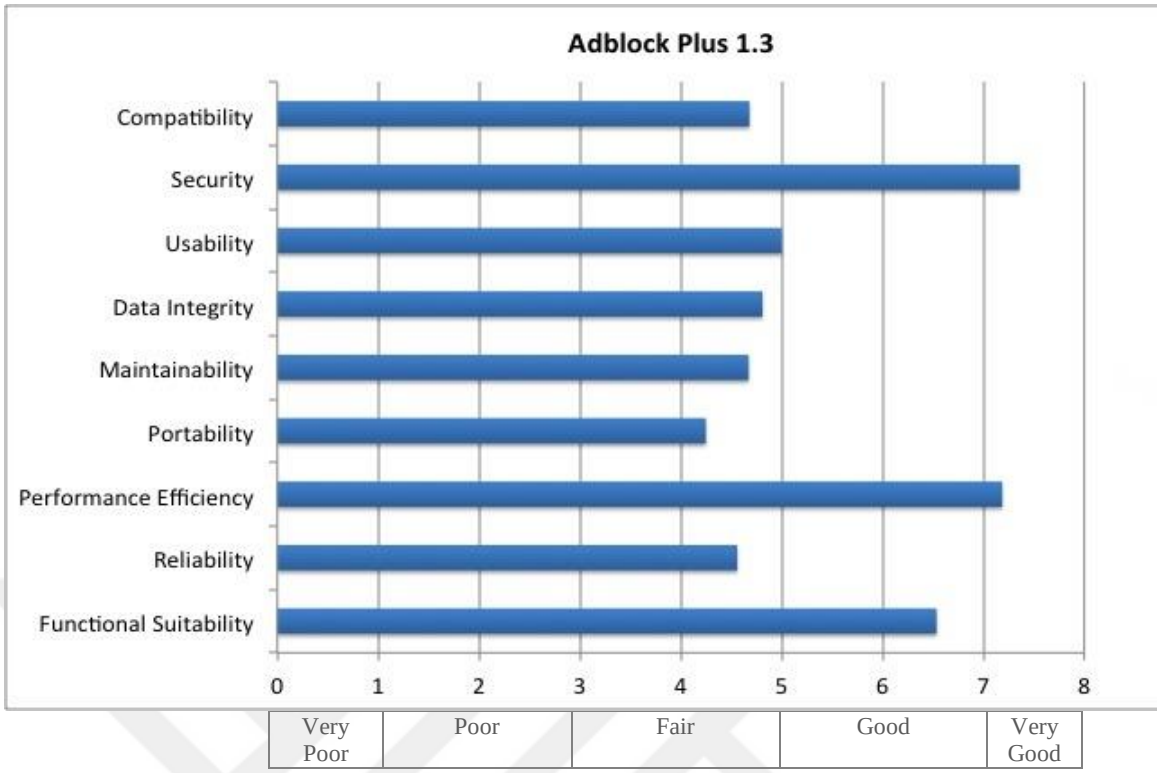


Figure 4. 6: Quality Scores of the Last Release of Adblock Plus

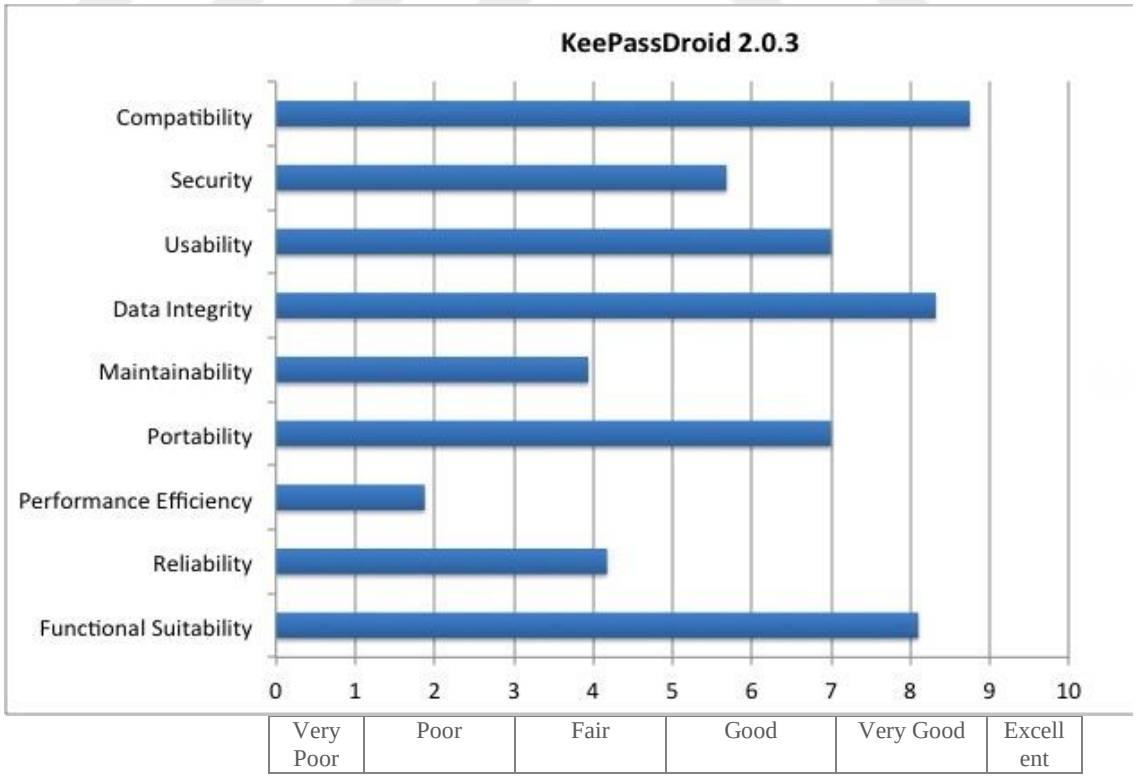


Figure 4. 7: Quality Scores of the Last Release of KeePassDroid

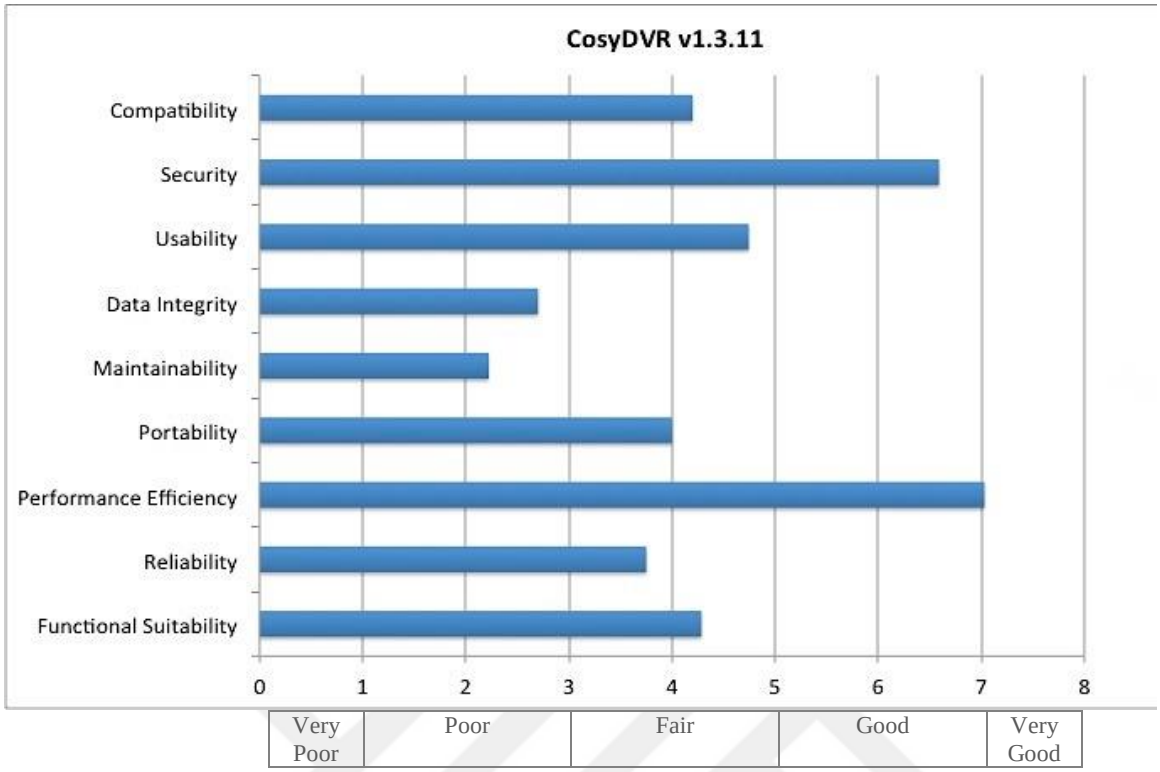


Figure 4. 8: Quality Scores of the Last Release of CosyDVR

The number of downloads and reviews of the Adblock Plus, KeepassDroid and CosyDVR applications are consistent with the quality scores of the applications evaluated by our Android Applications Quality Model. According to the reviews of users, star values of Adblock Plus and KeePassDroid are high; on the contrary star value of CosyDVR is low in Google Play Store. The number of downloads and the star values of the applications in Google Play Store are:

- Adblock Plus [60]: The number of downloads is in the range of 1,000,000 - 5,000,000. According to reviews of 41,503 users, the application has 4.1 stars out of 5.0.
- KeePassDroid [61]: The number of downloads is in the range of 1,000,000 - 5,000,000. According to reviews of 31,305 users, the application has 4.6 stars out of 5.0.
- CosyDVR [62]: The number of downloads is in the range of 100 - 500. According to reviews of 6 users, the application has 2.5 stars out of 5.0.

CHAPTER V

5 CONCLUSION AND FUTURE WORK

The increase in mobile devices causes the number of mobile applications to grow dramatically. As the number of mobile applications is increasing at a very high rate, quality of the applications has become an important issue. As quality is the composition of many characteristics, it is usually captured in a model that represents the quality characteristics and their relationships. Quality Models guide developers in the process of improving and maintaining the quality of their mobile applications. In this chapter of the thesis, the model development process concluded. We discuss the limitations encountered in our study in the second section. In third section, we present the future study directions.

5.1 CONCLUSION

In this thesis study, we proposed a quality model for Android applications. In the process of Android Applications Quality Model development, we have analyzed traditional quality characteristics, which are described in ISO/IEC SQuaRE Software Quality Standard and selected applicable quality characteristics. Afterwards, we have identified new Android-specific source code metrics and quality characteristics.

We have determined the relationship of source code metrics and quality characteristics by conducting a survey to Android Developers. We have developed our Android Applications Quality Model, validated our Quality Model and applied this model to different releases of three open-source Android applications in a case study to show the applicability of the model. We developed a general model for Android applications. Android developers can modify the quality model according to specific applications they prefer to apply.

Mobile software quality assessment is a recent discipline. In this study, we have started the assessment from the bottom by identifying the source code metrics, and developed a quality model for Android applications.

5.2 LIMITATIONS

There are some limitations of this study. First of all collecting data from mobile application developers was a painful process. As many developers do not prefer to participate in surveys, it was difficult to increase the number of participants. We emailed our survey to several IT Companies in Turkey and our personal connections. One of the threads to validity in this study is about the bias of selection. To mitigate this issue and include developers out of our contact network we emailed our survey to Android Developer Groups in Social Media (Facebook, Linkedin). 33 random

Android developers participated the survey. Secondly, reaching participants from different countries was difficult. We used social media to reach them, but convincing those developers to participate the survey was hard to achieve. As a result, all of the developers who participated the survey were from Turkey. Since the characteristics of Turkish developers could be different from those from other countries, the overall results may change when developers from other countries participate the survey. We plan to overcome this limitation in future work. Thirdly, all participants have the ability to develop Android applications but more than half of the participants developed one to four Android applications. It was difficult to include more developers who developed more than ten applications in the study. As Android was born on 2007 and started to become widespread on 2010 [63], finding participants who have been working on this domain for a long time was difficult. To mitigate this issue, we emailed our survey to all major mobile application development companies in Turkey. As a result, 5 out of 33 developers who developed more than ten applications participated the survey. Lastly, finding evaluators and including them in our study were hard to achieve. We overcame this limitation by inviting evaluators through the Android developers' network to include them in model validation process. We included three evaluators in our study.

5.3 FUTURE WORK

As a future work, some improvements will be made on this quality model. We plan to do a validated and statistically detailed analysis of source code metrics in the large scope of mobile applications. We plan to include different types of mobile platforms (e.g. IOS, Windows Mobile) in the quality model. We will perform the model development with the data gathered from more mobile application developers. We plan to reach more mobile application developers from different countries. In this manner, the results about the system will be more efficient. In addition, we will include more evaluators in our study to validate the quality model. The increase in evaluator number will enhance the reliability of the model.

REFERENCES

1. Yildirim, C. & Correia, A. Understanding Nomophobia: A Modern Age Phobia Among College Students, presented at Second International Conference, Held as Part of HCI International 2015, Los Angeles, 2015. Los Angeles, LA: Springer International Publishing.
2. Predicts 2014: Apps, Personal Cloud and Data Analytics Will Drive New Consumer Interactions. (n.d.). Retrieved October 21, 2015, from https://www.gartner.com/doc/2628016/predicts--apps-personal-cloud
3. Standards. (1990) IEEE Standard Glossary of Software Engineering Terminology. (Std 610.121990). Retrieved from http://ieeexplore.ieee.org/xpl/articleDetails.jsp?&arnumber=159342
4. The Statistics Portal: Number of Apps Available in Leading App Stores. (n.d.). Retrieved October 24, 2015, from http://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/
5. BrainyQuote: Galileo Galilei Quotes. (n.d.). Retrieved October 28, 2015, from http://www.brainyquote.com/quotes/quotes/g/galileogal381325.html
6. Robson, C. (2002). Real world research: a resource for social scientists and practitioner-researchers. Retrieved from http://www.dem.fmed.uc.pt/Bibliografia/Livros_Educacao_Medica/Livro34.pdf
7. Malhotra, N. & Pruthi, S. (2012). An Efficient Software Quality Models for Safety and Resilience. International Journal of Recent Technology and Engineering (IJRTE). 1 (3). 66-70
8. Quality Models in Software Engineering. (n.d.). Retrieved November 3, 2015, from https://msritse2012.wordpress.com/2013/01/27/quality-models-in-software-engineering/
9. McCall Quality Model. (n.d.). Retrieved November 8, 2015, from http://maisqual.squoring.com/wiki/index.php/McCall_Quality_Model
10. Boehm Quality Model. (n.d.). Retrieved November 8, 2015, from http://maisqual.squoring.com/wiki/index.php/Boehm_Quality_Model
11. Samadhiya, D., Wang, S. & Chen, D. Quality Models: Role and Value in Software Engineering, presented at 2nd International Conference on Software Technology and Engineering (ICSTE), San Juan, PR, 2010. IEEE.
12. Suman. & Wadhwa, M. (2014). A Comparative Study of Software Quality Models. International Journal of Computer Science and Information Technologies (IJCSIT). 5 (4). 5634-5638
13. Wagner, S. (2013). Software Product Quality Control. Retrieved from http://www.springer.com/it/book/9783642385704

14. Maryoly, O., M.A. Perez & T. Rojas. (2006). A systemic quality model for evaluating software products. Laboratorio de Investigcin en Sistemas de Informacin. 2 (4). 373-381
15. ISO/IEC 9126-1:2000 (2000). International Standard. "Information technology- Software product quality - Part 1: Quality model". (Std. FDIS 9126-1) Retrieved from http://www.cse.unsw.edu.au/~cs3710/PMmaterials/Resources/9126-1%20Standard.pdf
16. Tripathi, S. (2014). A Survey on Quality Perspective and Software Quality Models. IOSR Journal of Computer Engineering (IOSR-JCE). 16 (2). 63-72
17. Bansiya, J. & Davis, C. A (2002) Hierarchical Model for Object-Oriented Design Quality Assessment. IEEE Transactions on software engineering. 28 (1). 4-17
18. ISO/IEC 25010:2011. (n.d.). Retrieved December 2, 2015, from https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en
19. Franke, D. & Kowalewski, S. A Mobile Software Quality Model, presented at 12th International Conference on Quality Software, Xi'an, Shaanxi, 2012. IEEE.
20. Franke, D. & Weise, C. Providing a Software Quality Framework for Testing of Mobile Applications, presented at Fourth IEEE International Conference on Software Testing, Verification and Validation, Berlin, 2011. IEEE.
21. Idri, A., Moumane, K. & Abran, A. On the Use of Software Quality Standard ISO / IEC 9126 in Mobile Environments, presented at 20th Asia-Pacific Software Engineering Conference, Aveiro, 2013. IEEE.
22. Zahra, S., Khalid, A. & Javed, A. (2013). An Efficient and Effective New Generation Objective Quality Model for Mobile Applications. I.J.Modern Education and Computer Science (MECS). 4. 36-42
23. Jost, G., Huber, J., & Hericko. M. Using Object Oriented Software Metrics for Mobile Application Development, presented at 2nd Workshop of Software Quality Analysis, Monitoring, Improvement and Applications (SQAMIA), Serbia, 2013. Published at http://ceur-ws.org.
24. Hecht, G., Benomar, O., Rouvoy, R., Moha, N. & Duchien, L. Tracking the Software Quality of Android Applications along their Evolution, presented at 30th IEEE/ACM International Conference on Automated Software Engineering, United States, 2015. IEEE.
25. Android Developer Page. (n.d.). Retrieved November 12, 2015, from http://developer.android.com/
26. Hoberg, J. (2013). Testing Mobile Applications: A Model for Mobile Testing. Retrieved from http://www.slideshare.net/JohanHoberg/testing-mobile-applications-29290125
27. TechCrunch Research. (n.d.). Retrieved December 10, 2015, from http://techcrunch.com/2013/03/12/users-have-low-tolerance-for-buggy-apps-only-16-will-try-a-failing-app-more-than-twice/
28. Ferguson, D. (2002). Mobile .NET. Retrieved from https://books.google.com.tr/books?id=fYInCgAAQBAJ
29. Lee, W. (2015, December 10). Mobile Apps and Power Consumption Basics, Part 1 [Web log post]. Retrieved from https://developer.qualcomm.com/blog/mobile-apps-and-power-consumption-basics-part-1

30. IDC. Smartphone OS Market Share, 2015 Q2. (n.d.). Retrieved December 12, 2015, from http://www.idc.com/prodserv/smartphone-os-market-share.jsp

31. Category: Android (Operating System) Devices. (n.d.). Retrieved November 12, 2015, from https://en.wikipedia.org/w/index.php?title=Category:Android_(operating_system)_devices&pagefrom=Xiaomi+Mi+2A#mw-pages

32. Harrison, R., Flood, D. & Duce, D. (2013). Usability of mobile applications: literature review and rationale for a new usability model. Journal of Interaction Science. 1(1). 1-16

33. Arxan Research: State of the Application Security. (n.d.). Retrieved December 15, 2015, from https://www.arxan.com/resources/state-of-application-security/

34. Android Developer Page: Bound Services. (n.d.). Retrieved December 15, 2015, from http://developer.android.com/guide/components/bound-services.html

35. Jetter, A. (2006). Assessing Software Quality Attributes with Source Code Metrics. (Diploma Thesis). Retrieved from http://www.ifi.uzh.ch/seal/research/tools/archive/swQuality/da_jetter.pdf

36. Meier, R. (2012). Professional Android 4 Application Development. Retrieved from https://books.google.com.tr/books?hl=en&lr=&id=bmJIl_wPgQsC

37. Smyth, N. (2014). Android Studio Development Essentials. Retrieved from http://www.techotopia.com/index.php/Android_Studio_Development_Essentials

38. Smyth, N. (2014). Android 4 App Development Essentials. Retrieved from http://www.techotopia.com/index.php/Android_4_App_Development_Essentials

39. Murphy, M. (2012). The Busy Coder's Guide to Android Development. Retrieved from https://commonsware.com/Android/Android_3-6-CC.pdf

40. Gramlich, N. (2012). Android Programming with Tutorials from the anddev.org-Community. Retrieved from http://andbook.anddev.org/files/andbook.pdf

41. Önder, M. & MERMERKAYA, A. (2013). Merhaba Android. İstanbul: Deniz Ofset

42. Vogel, L. (2015) Introduction to Android development with Android Studio. Retrieved from http://www.vogella.com/tutorials/Android/article.html

43. Android Development. (n.d.). Retrieved November 20, 2015, from https://teamtreehouse.com/tracks/android-development

44. Developing Android Apps Android Fundamentals. (n.d.). Retrieved November 30, 2015, from https://www.udacity.com/course/developing-android-apps--ud853

45. Difference between Android and Java. (n.d.). Retrieved December 17, 2015, from http://www.differencebetween.com/difference-between-android-and-vs-java/#ixzz2TIkUX7cI

46. Software Package Metrics. (n.d.). Retrieved December 20, 2015, from https://en.wikipedia.org/wiki/Software_package_metrics

47. Google Forms. (n.d.). Retrieved December 20, 2015, from https://www.google.com/forms/about/

48. George, D., & Mallery, P. (2003). SPSS for Windows step by step: A simple guide and reference. Retrieved from http://wps.ablongman.com/wps/media/objects/385/394732/george4answers.pdf

49. 2048-android game in Google Play Store. (n.d.). Retrieved December 25, 2015, from https://play.google.com/store/apps/details?id=com.gabrielecirulli.app2048

50. IBM SPSS Statistics. (n.d.). Retrieved December 20, 2015, from http://www-01.ibm.com/software/analytics/spss/

51. F-Droid Limited. (n.d.). Retrieved December 22, 2015, from https://f-droid.org/wiki/index.php?title=Category:Apps.

52. Feature Scaling (n.d.). Retrieved December 15, 2015, from https://en.wikipedia.org/wiki/Normalization_(statistics)

53. Survey Methods: When is it Generally Better to Use Median over Mean? (n.d.). Retrieved December 20, 2015, from http://blog.surveymethods.com/when-is-it-generally-better-to-use-median-over-mean/

54. Paired Sample T-Tests. (n.d.). Retrieved December 26, 2015, from http://libguides.library.kent.edu/SPSS/PairedSamplestTest

55. Adblock Plus. (n.d.). Retrieved December 22, 2015, from https://adblockplus.org/

56. KeePassDroid. (n.d.). Retrieved December 22, 2015, from http://www.keepassdroid.com/

57. CosyDVR. (n.d.). Retrieved February 10, 2015 from https://github.com/sergstetsuk/CosyDVR

58. Eclipse Metrics Plugin 1.3.8. (n.d.). Retrieved December 25, 2015, from http://metrics2.sourceforge.net/

59. Data Normalization and Standardization. (n.d.). Retrieved December 25, 2015, from http://www.benetzkorn.com/2011/11/data-normalization-and-standardization/

60. Adblock Browser for Android in Google Play Store. (n.d.). Retrieved December 25, 2015, from https://play.google.com/store/apps/details?id=org.adblockplus.browser&hl=en

61. KeePassDroid in Google Play Store. (n.d.). Retrieved December 25, 2015, from https://play.google.com/store/apps/details?id=com.android.keepass&hl=en

62. CosyDVR in Google Play Store. (n.d.). Retrieved February 10, 2015, from https://play.google.com/store/apps/details?id=es.esy.CosyDVR

63. The Android Story. (n.d.). Retrieved February 10, 2015, from http://www.xcubelabs.com/infographic-android-story/

APPENDICES

Appendix A: Survey Questions

Android Applications Quality Model Survey

*Required

1. **1- How many years of work experience do you have in IT and software development industries?** *

   Mark only one oval.

   - ◯ 0
   - ◯ 1-4
   - ◯ 5-10
   - ◯ 10+

2. **2- How many Android Projects are you involved in development?** *

   Mark only one oval.

   - ◯ 0
   - ◯ 1-4
   - ◯ 5-10
   - ◯ 10+

3. **3- Do you develop software for your company or for yourself?** *

   Mark only one oval.

   - ◯ Company
   - ◯ Myself
   - ◯ Both

Explanation About Following Questions

We are investigating the effect of ISO SQuaRE Quality Attributes on the quality of Android Mobile Applications. The Quality Attributes are Functional Suitability, Reliability, Performance Efficiency, Portability, Maintainability, Data Integrity/Persistance, Usability, Security, Compatibility.

For each of the Quality Attribute in the questions below, please indicate the effect of factors. (Negative, Positive, No Effect)

4. **4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.** *

FUNCTIONAL SUITABILITY: Degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

5. **What other factors would you offer that affects FUNCTIONAL SUITABILITY?**

.....

6. **5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.** *

RELIABILITY: Degree to which a system or component performs specified functions under specified conditions for a specified period of time.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

7. **What other factors would you offer that affects RELIABILITY?**

.....

8. **6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.** *

PERFORMANCE EFFICIENCY: The performance relative to the amount of resources used under stated conditions.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

9. **What other factors would you offer that affects PERFORMANCE EFFICIENCY?**

.....

10. **7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.** *

PORTABILITY: Degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

11. **What other factors would you offer that affects PORTABILITY?**

.....

12. **8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.** *

MAINTAINABILITY: Degree of effectiveness and efficiency with the product can be modified.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

13. **What other factors would you offer that affects MAINTAINABILITY?**

.....

14. **9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.** *

DATA INTEGRITY: Ability to keep information even after mobile application is paused or killed.

Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

15. **What other factors would you offer that affects DATA INTEGRITY?**

.....

16. **10- Select the degree of effects for each factors on the USABILITY of Mobile Application.** *

USABILITY: Degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

17. **What other factors would you offer that affects USABILITY?**

.....

18. **11- Select the degree of effects for each factors on the SECURITY of Mobile Application.** *

SECURITY: Degree to which a product or system protects information and data so that persons or other products or systems have the degree of data access appropriate to their types and levels of authorization.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

19. **What other factors would you offer that affects SECURITY?**

.....

20. **12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.** *

COMPATIBILITY: Degree to which a product, system or component can exchange information with other products, systems, or components, and/or perform its required functions, while sharing the same hardware or software environment.
Mark only one oval per row.

	Strongly Negative	Somewhat Negative	No Effect	Somewhat Positive	Strongly Positive
↑ Num of Classes	◯	◯	◯	◯	◯
↑ Num of Activities	◯	◯	◯	◯	◯
↑ Num of Services	◯	◯	◯	◯	◯
↑ Num of BroadcastReceivers	◯	◯	◯	◯	◯
↑ Num of ContentProviders	◯	◯	◯	◯	◯
↑ Number of Dialogs	◯	◯	◯	◯	◯
↑ Number of Threads	◯	◯	◯	◯	◯
↑ android:minSdkVersion	◯	◯	◯	◯	◯
↑ android:targetSdkVersion	◯	◯	◯	◯	◯
↑ Num of Tables in Database	◯	◯	◯	◯	◯
↑ Depth of Inheritance Tree	◯	◯	◯	◯	◯
↑ Degree of Coupling	◯	◯	◯	◯	◯
↑ Degree of Cohesion	◯	◯	◯	◯	◯
↑ Num of Attributes	◯	◯	◯	◯	◯
↑ Number of Methods	◯	◯	◯	◯	◯
↑ Number of Intents	◯	◯	◯	◯	◯
↑ Cyclomatic Complexity	◯	◯	◯	◯	◯
↑ Num of <uses-permission>	◯	◯	◯	◯	◯

21. **What other factors would you offer that affects COMPATIBILITY?**

.....

Appendix B: Ethics Approval Form

UYGULAMALI ETİK ARAŞTIRMA MERKEZİ
APPLIED ETHICS RESEARCH CENTER

ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

DUMLUPINAR BULVARI 06800
ÇANKAYA ANKARA/TURKEY
T: +90 312 210 22 91
F: +90 312 210 79 59
ueam@metu.edu.tr
www.ueam.metu.edu.tr

Sayı: 28620816 /249

21 ARALIK 2015

Gönderilen: Doç.Dr. Aysu BETİN CAN
Bilişim Sistemleri

Gönderen: Prof. Dr. Canan SÜMER
İnsan Araştırmaları Komisyonu Başkanı

İlgi: Etik Onayı

Sayın Doç.Dr. Aysu BETİN CAN danışmanlığını yaptığınız Merve Vildan ŞİMŞEK'in "Android Uygulamalarının Kalite Özelliklerinin Belirlenmesi isimli araştırması İnsan Araştırmaları Komisyonu tarafından uygun görülerek gerekli onay 01.01.2015-01.06.2016 tarihleri arasında geçerli olmak üzere verilmiştir.

Prof. Dr. Canan SÜMER
Uygulamalı Etik Araştırma Merkezi
İnsan Araştırmaları Komisyonu Başkanı

BU BÖLÜM, ILGILI BÖLÜMLERİ TEMSİL EDEN İNSAN ARAŞTIRMALARI ETİK ALT KURULU TARAFINDAN DOLDURULACAKTIR.

Protokol No: 2015-FEN-071

İAEK DEĞERLENDİRME SONUCU

Sayın Hakem,

Aşağıda yer alan üç seçenekten birini işaretleyerek değerlendirmenizi tamamlayınız. Lütfen "Revizyon Gereklidir" ve "Ret" değerlendirmeleri için gerekli açıklamaları yapınız.

Değerlendirme Tarihi: 19.12.2015

Ad Soyad:

☒ **Herhangi bir değişikliğe gerek yoktur. Veri toplama/uygulama başlatılabilir.**

☐ **Revizyon gereklidir**

☐ Gönüllü Katılım Formu yoktur.

☐ Gönüllü Katılım Formu eksiktir.

Gerekçenizi ayrıntılı olarak açıklayınız:

☐ Katılım Sonrası Bilgilendirme Formu yoktur.

☐ Katılım Sonrası Bilgilendirme Formu eksiktir.

Gerekçenizi ayrıntılı olarak açıklayınız:

☐ Rahatsızlık kaynağı olabilecek sorular/maddeler ya da prosedürler içerilmektedir.

Gerekçenizi ayrıntılı olarak açıklayınız:

☐ Diğer.

Gerekçenizi ayrıntılı olarak açıklayınız:

☐ **Ret**

Ret gerekçenizi ayrıntılı olarak açıklayınız:

8

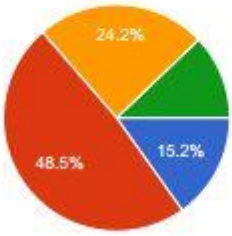
Appendix C: Summary of Survey Results

33 responses

View all responses Publish analytics

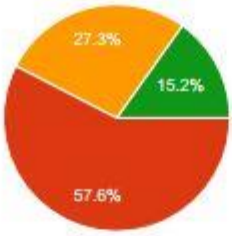
Summary

1- How many years of work experience do you have in IT and software development industries?



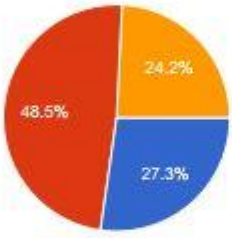
0	5	15.2%
1-4	16	48.5%
5-10	8	24.2%
10+	4	12.1%

2- How many Android Projects are you involved in development?



0	0	0%
1-4	19	57.6%
5-10	9	27.3%
10+	5	15.2%

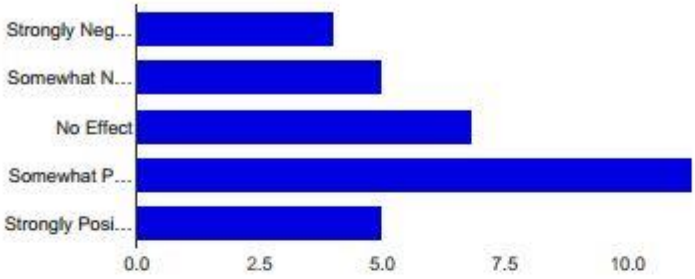
3- Do you develop software for your company or for yourself?



Company	9	27.3%
Myself	16	48.5%
Both	8	24.2%

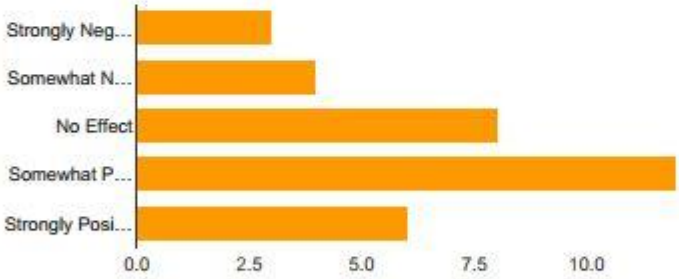
Explanation About Following Questions

↑ Num of Classes [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



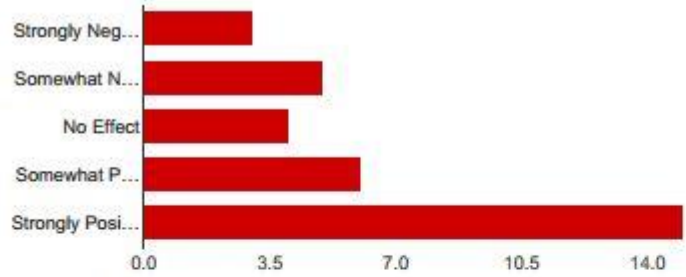
Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	7	21.2%
Somewhat Positive	12	36.3%
Strongly Positive	5	15.2%

↑ Num of Activities [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



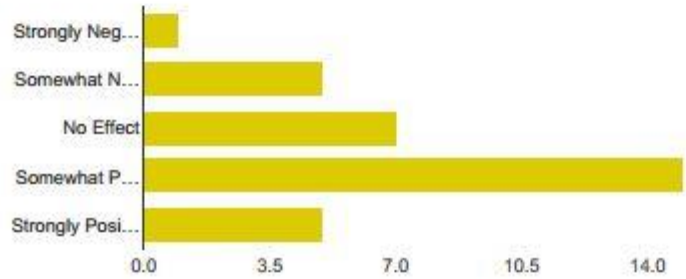
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	8	24.2%
Somewhat Positive	12	36.4%
Strongly Positive	6	18.2%

↑ Num of Services [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



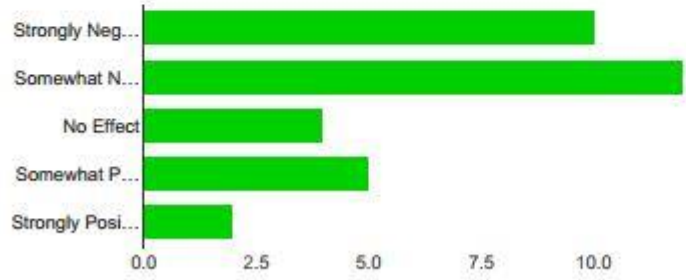
Strongly Negative	3	9.1%
Somewhat Negative	5	15.2%
No Effect	4	12.1%
Somewhat Positive	6	18.2%
Strongly Positive	15	45.5%

↑ Num of BroadcastReceivers [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



Strongly Negative	1	3%
Somewhat Negative	5	15.2%
No Effect	7	21.2%
Somewhat Positive	15	45.5%
Strongly Positive	5	15.2%

↑ Num of ContentProviders [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]

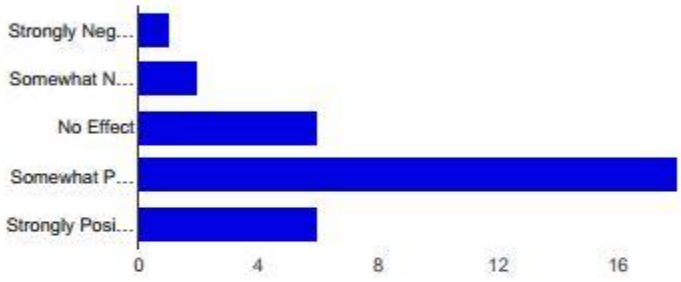


Strongly Negative	10	30.3%
Somewhat Negative	12	36.4%
No Effect	4	12.1%
Somewhat Positive	5	15.2%
Strongly Positive	2	6.1%

↑ Number of Dialogs [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]

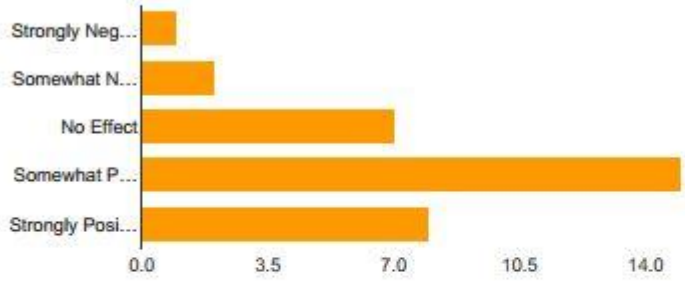
Strongly Negative	4	12.1%
Somewhat Negative	6	18.2%
No Effect	16	48.5%
Somewhat Positive	4	12.1%
Strongly Positive	3	9.1%

↑ Number of Threads [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



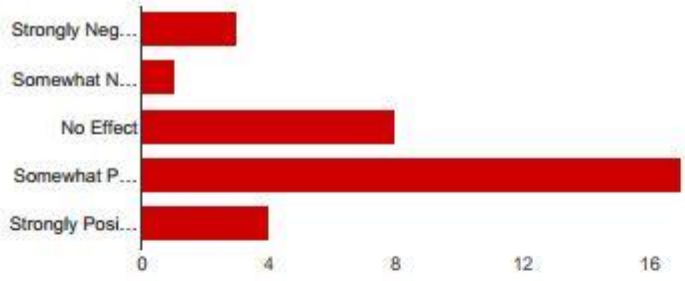
Strongly Negative	1	3%
Somewhat Negative	2	6.1%
No Effect	6	18.2%
Somewhat Positive	18	54.5%
Strongly Positive	6	18.2%

↑ android:minSdkVersion [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



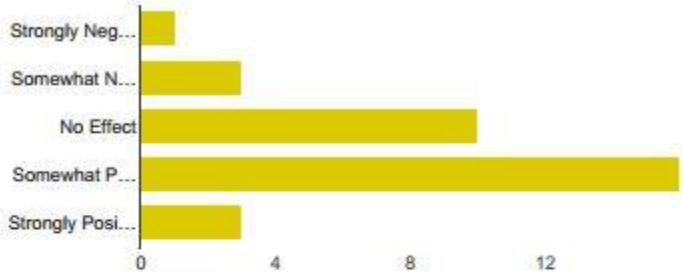
Strongly Negative	1	3%
Somewhat Negative	2	6.1%
No Effect	7	21.2%
Somewhat Positive	15	45.5%
Strongly Positive	8	24.2%

↑ android:targetSdkVersion [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]

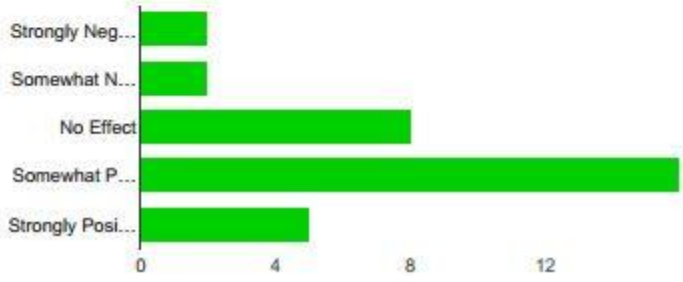


Strongly Negative	3	9.1%
Somewhat Negative	1	3%
No Effect	8	24.2%
Somewhat Positive	17	51.5%
Strongly Positive	4	12.1%

↑ Num of Tables in Database [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]

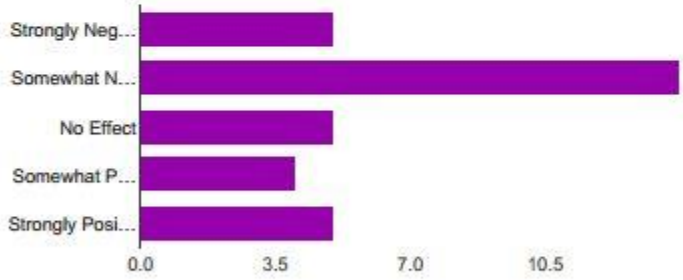


↑ Depth of Inheritance Tree [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



Strongly Negative	**2**	6.1%
Somewhat Negative	**2**	6.1%
No Effect	**8**	24.2%
Somewhat Positive	**16**	48.5%
Strongly Positive	**5**	15.2%

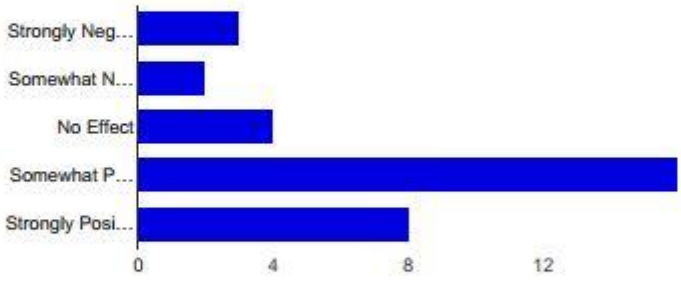
↑ Degree of Coupling [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



Strongly Negative	**5**	15.2%
Somewhat Negative	**14**	42.4%

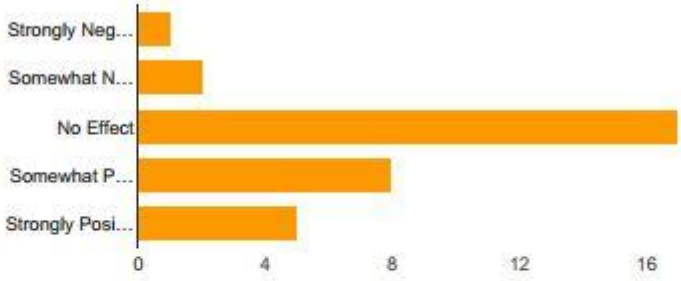
No Effect	5	15.2%
Somewhat Positive	4	12.1%
Strongly Positive	5	15.2%

↑ Degree of Cohesion [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



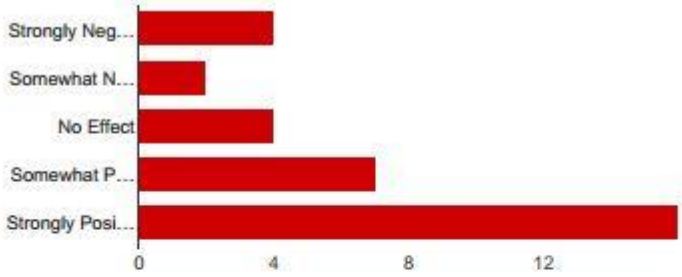
Strongly Negative	3	9.1%
Somewhat Negative	2	6.1%
No Effect	4	12.1%
Somewhat Positive	16	48.5%
Strongly Positive	8	24.2%

↑ Num of Attributes [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



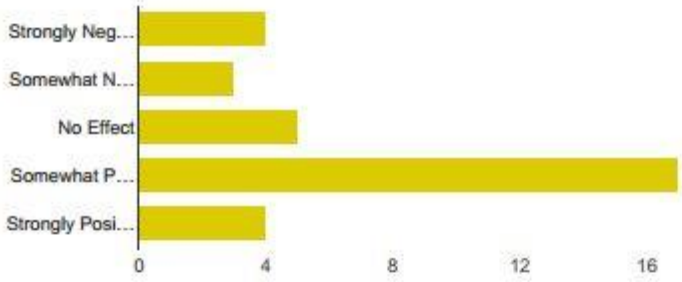
Strongly Negative	1	3%
Somewhat Negative	2	6.1%
No Effect	17	51.5%
Somewhat Positive	8	24.2%
Strongly Positive	5	15.2%

↑ Number of Methods [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



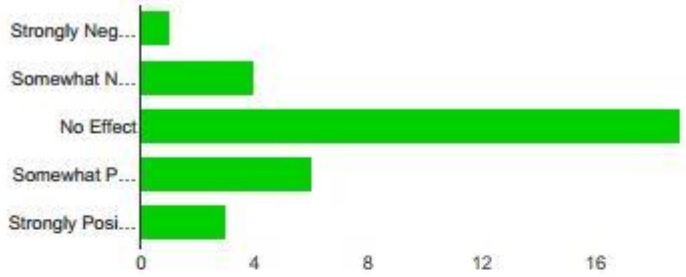
Strongly Negative	4	12.1%
Somewhat Negative	2	6.1%
No Effect	4	12.1%
Somewhat Positive	7	21.2%
Strongly Positive	16	48.5%

↑ Number of Intents [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]

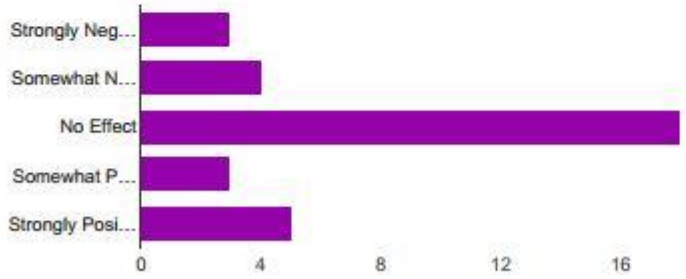


Strongly Negative	4	12.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	17	51.5%
Strongly Positive	4	12.1%

↑ Cyclomatic Complexity [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]



↑ Num of <uses-permission> [4- Select the degree of effects for each factors on the FUNCTIONAL SUITABILITY of Mobile Application.]

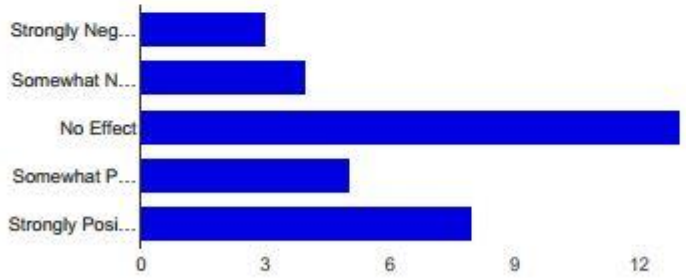


Strongly Negative	**3**	9.1%
Somewhat Negative	**4**	12.1%
No Effect	**18**	54.5%
Somewhat Positive	**3**	9.1%
Strongly Positive	**5**	15.2%

What other factors would you offer that affects FUNCTIONAL SUITABILITY?

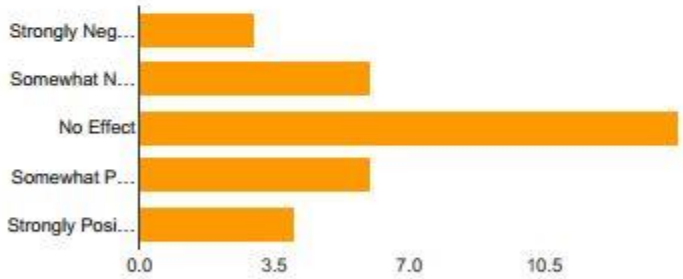
Utility

↑ Num of Classes [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



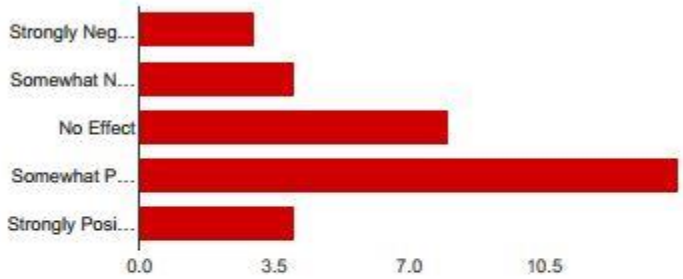
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	13	39.4%
Somewhat Positive	5	15.2%
Strongly Positive	8	24.2%

↑ Num of Activities [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



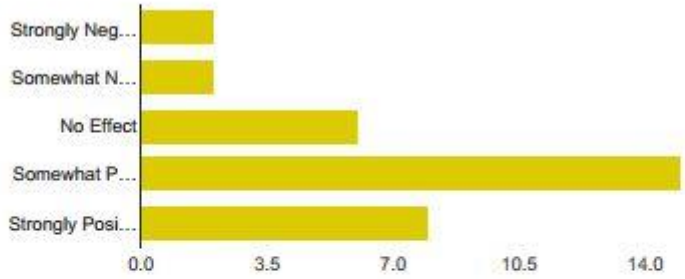
Strongly Negative	3	9.1%
Somewhat Negative	6	18.2%
No Effect	14	42.4%
Somewhat Positive	6	18.2%
Strongly Positive	4	12.1%

↑ Num of Services [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



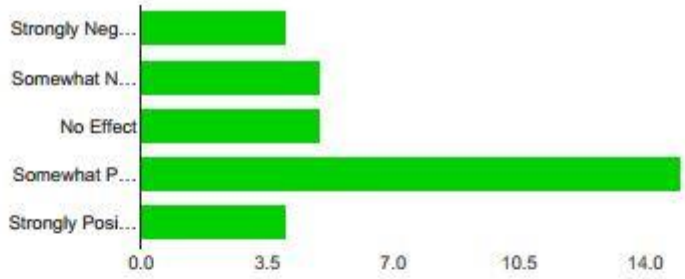
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	8	24.2%
Somewhat Positive	14	42.4%
Strongly Positive	4	12.1%

↑ Num of BroadcastReceivers [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



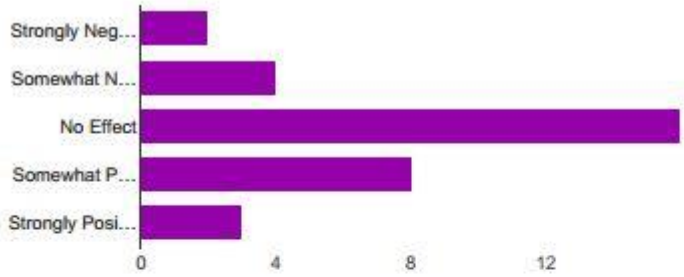
Strongly Negative	**2**	6.1%
Somewhat Negative	**2**	6.1%
No Effect	**6**	18.2%
Somewhat Positive	**15**	45.5%
Strongly Positive	**8**	24.2%

↑ Num of ContentProviders [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]

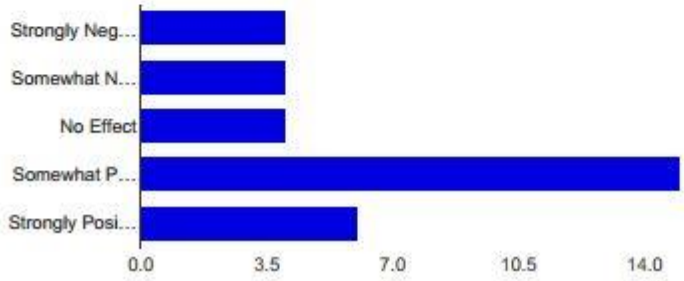


Strongly Negative	**4**	12.1%
Somewhat Negative	**5**	15.2%
No Effect	**5**	15.2%
Somewhat Positive	**15**	45.5%
Strongly Positive	**4**	12.1%

↑ Number of Dialogs [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]

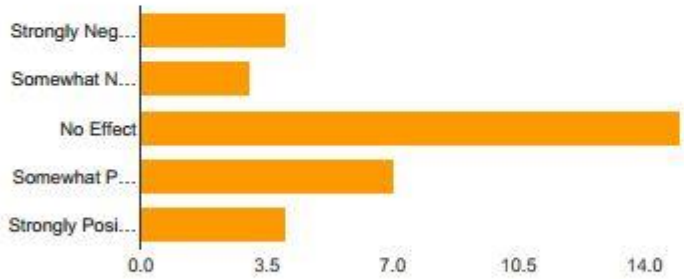


↑ Number of Threads [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



Strongly Negative	**4**	12.1%
Somewhat Negative	**4**	12.1%
No Effect	**4**	12.1%
Somewhat Positive	**15**	45.5%
Strongly Positive	**6**	18.2%

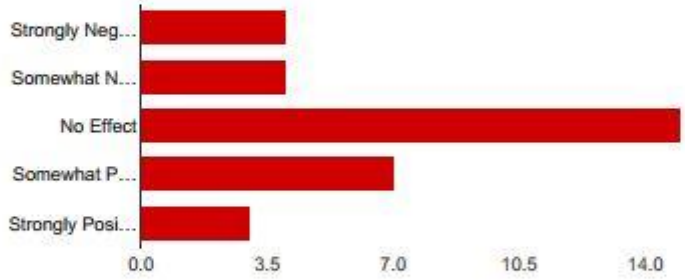
↑ android:minSdkVersion [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



Strongly Negative	**4**	12.1%
Somewhat Negative	**3**	9.1%

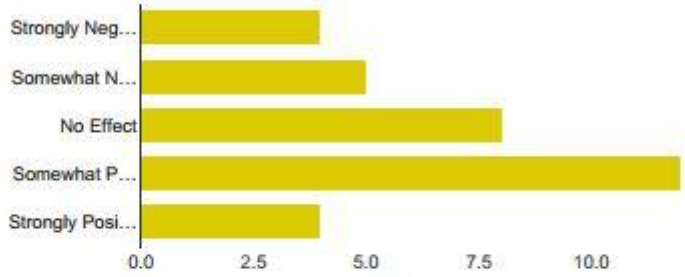
No Effect	15	45.5%
Somewhat Positive	7	21.2%
Strongly Positive	4	12.1%

↑ android:targetSdkVersion [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



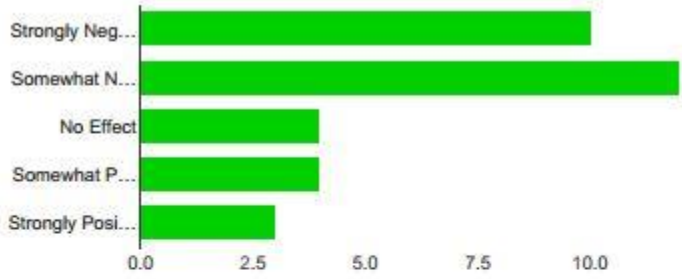
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	15	45.5%
Somewhat Positive	7	21.2%
Strongly Positive	3	9.1%

↑ Num of Tables in Database [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



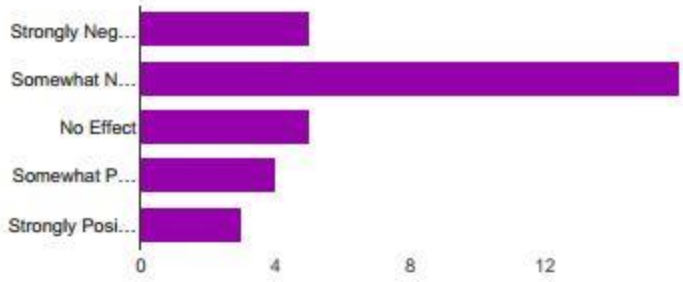
Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	8	24.2%
Somewhat Positive	12	36.4%
Strongly Positive	4	12.1%

↑ Depth of Inheritance Tree [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



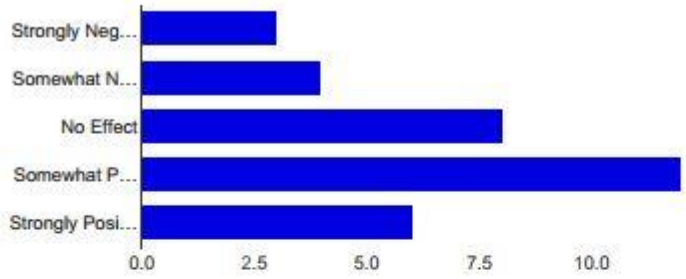
Strongly Negative	**10**	30.3%
Somewhat Negative	**12**	36.4%
No Effect	**4**	12.1%
Somewhat Positive	**4**	12.1%
Strongly Positive	**3**	9.1%

↑ Degree of Coupling [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]

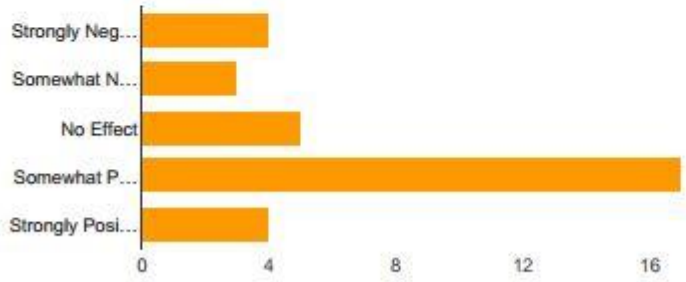


Strongly Negative	**5**	15.2%
Somewhat Negative	**16**	48.5%
No Effect	**5**	15.2%
Somewhat Positive	**4**	12.1%
Strongly Positive	**3**	9.1%

↑ Degree of Cohesion [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]

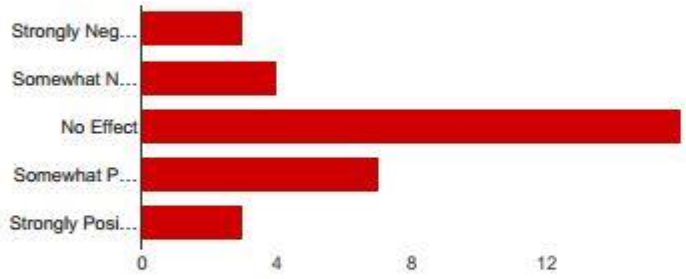


↑ Num of Attributes [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



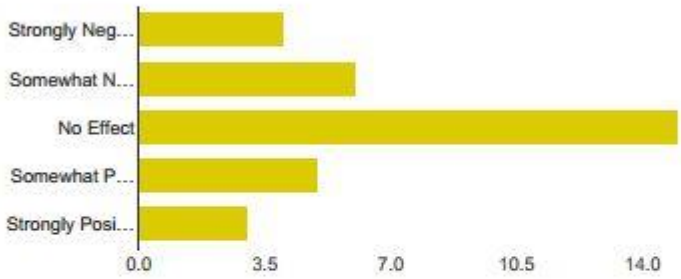
Strongly Negative	**4**	12.1%
Somewhat Negative	**3**	9.1%
No Effect	**5**	15.2%
Somewhat Positive	**17**	51.5%
Strongly Positive	**4**	12.1%

↑ Number of Methods [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



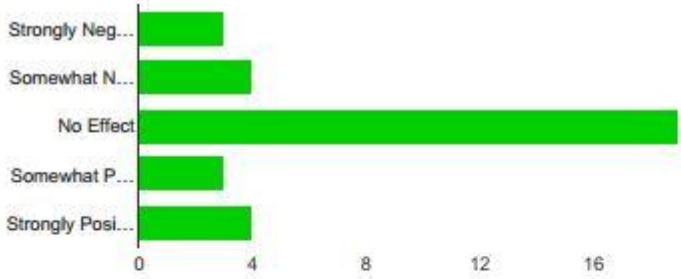
Strongly Negative	**3**	9.1%
Somewhat Negative	**4**	12.1%
No Effect	**16**	48.5%
Somewhat Positive	**7**	21.2%
Strongly Positive	**3**	9.1%

↑ Number of Intents [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



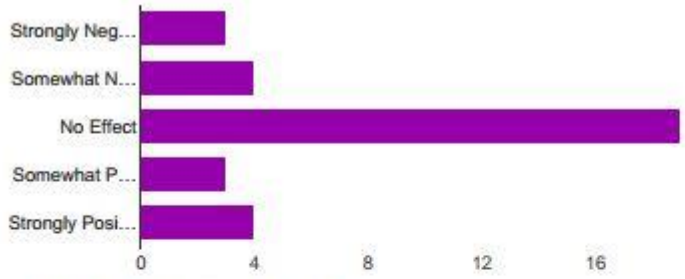
Strongly Negative	**4**	12.1%
Somewhat Negative	**6**	18.2%
No Effect	**15**	45.5%
Somewhat Positive	**5**	15.2%
Strongly Positive	**3**	9.1%

↑ Cyclomatic Complexity [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



Strongly Negative	**3**	9.1%
Somewhat Negative	**4**	12.1%
No Effect	**19**	57.6%
Somewhat Positive	**3**	9.1%
Strongly Positive	**4**	12.1%

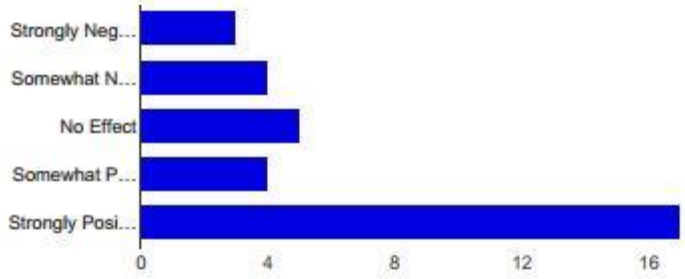
↑ Num of <uses-permission> [5- Select the degree of effects for each factors on the RELIABILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	19	57.6%
Somewhat Positive	3	9.1%
Strongly Positive	4	12.1%

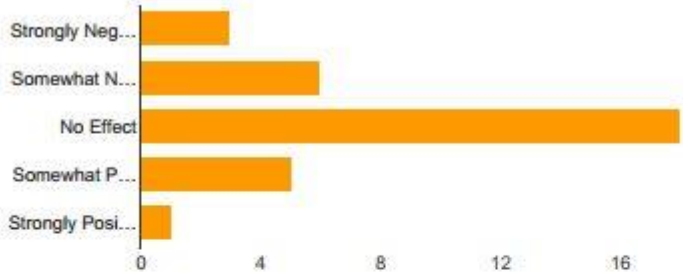
What other factors would you offer that affects RELIABILITY?

↑ Num of Classes [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]

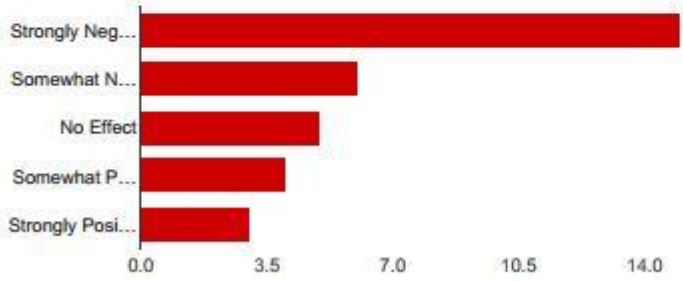


Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	5	15.2%
Somewhat Positive	4	12.1%
Strongly Positive	17	51.5%

↑ Num of Activities [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]

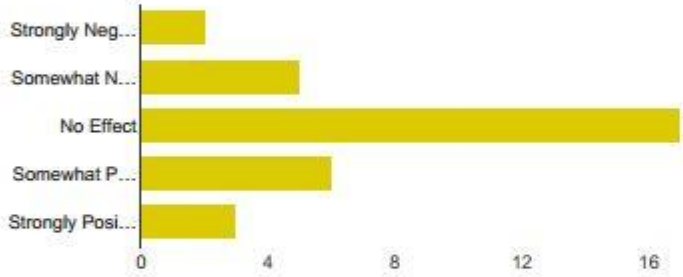


↑ Num of Services [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



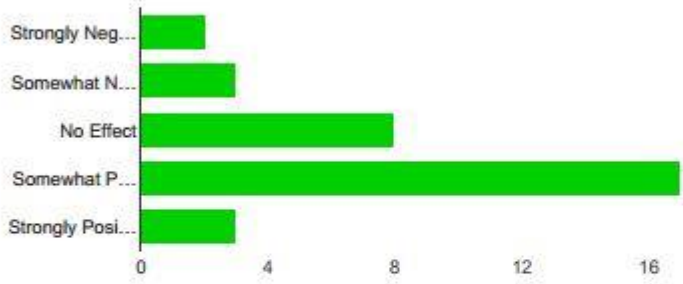
Strongly Negative	**15**	45.5%
Somewhat Negative	**6**	18.2%
No Effect	**5**	15.2%
Somewhat Positive	**4**	12.1%
Strongly Positive	**3**	9.1%

↑ Num of BroadcastReceivers [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



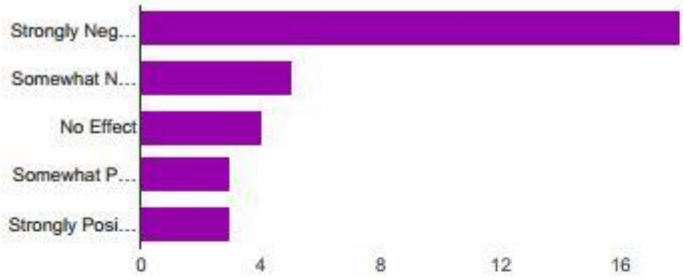
Strongly Negative	**2**	6.1%
Somewhat Negative	**5**	15.2%
No Effect	**17**	51.5%
Somewhat Positive	**6**	18.2%
Strongly Positive	**3**	9.1%

↑ Num of ContentProviders [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



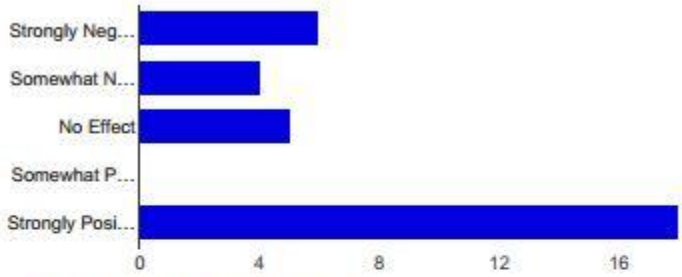
Strongly Negative	2	6.1%
Somewhat Negative	3	9.1%
No Effect	8	24.2%
Somewhat Positive	17	51.5%
Strongly Positive	3	9.1%

↑ Number of Dialogs [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



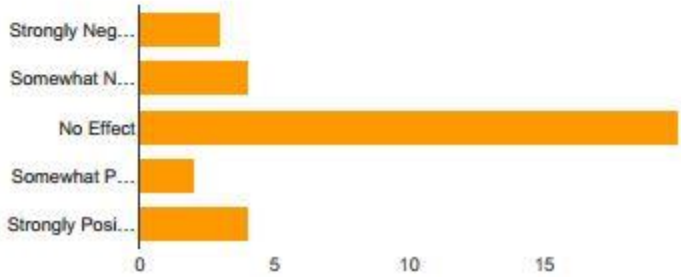
Strongly Negative	18	54.5%
Somewhat Negative	5	15.2%
No Effect	4	12.1%
Somewhat Positive	3	9.1%
Strongly Positive	3	9.1%

↑ Number of Threads [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



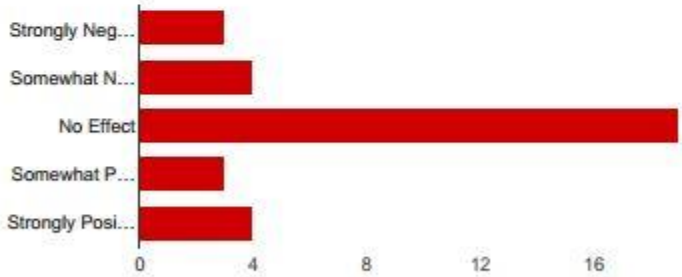
Strongly Negative	6	18.2%
Somewhat Negative	4	12.1%
No Effect	5	15.2%
Somewhat Positive	0	0%
Strongly Positive	18	54.5%

↑ android:minSdkVersion [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	20	60.6%
Somewhat Positive	2	6.1%
Strongly Positive	4	12.1%

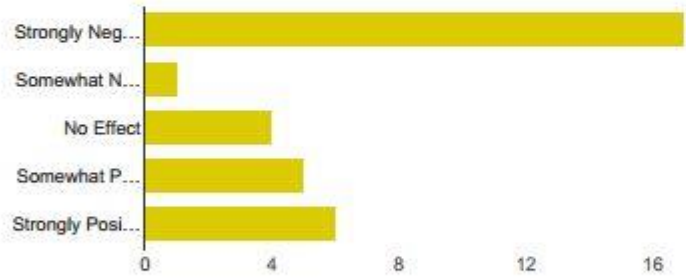
↑ android:targetSdkVersion [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



Strongly Negative	3	9.1%

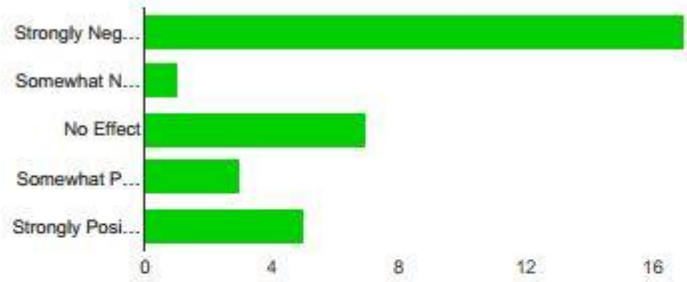
Somewhat Negative	4	12.1%
No Effect	19	57.6%
Somewhat Positive	3	9.1%
Strongly Positive	4	12.1%

↑ Num of Tables in Database [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



Strongly Negative	17	51.5%
Somewhat Negative	1	3%
No Effect	4	12.1%
Somewhat Positive	5	15.2%
Strongly Positive	6	18.2%

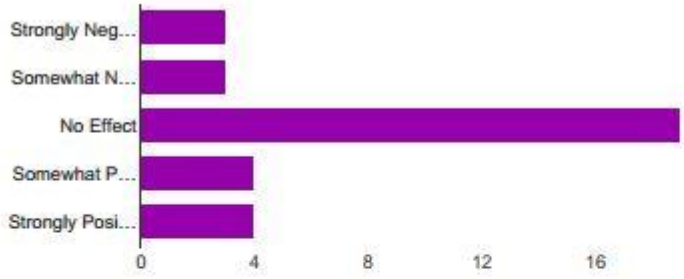
↑ Depth of Inheritance Tree [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



Strongly Negative	17	51.5%
Somewhat Negative	1	3%
No Effect	7	21.2%
Somewhat Positive	3	9.1%
Strongly Positive	5	15.2%

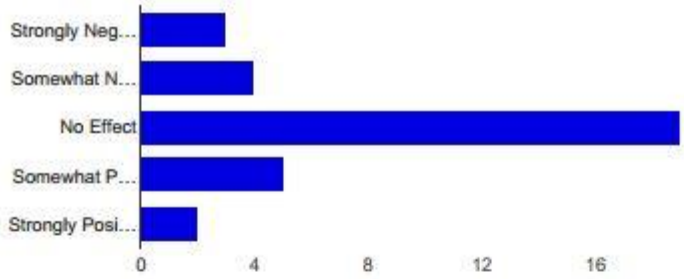
↑ Degree of Coupling [6- Select the degree of effects for each factors on

the PERFORMANCE EFFICIENCY of Mobile Application.]



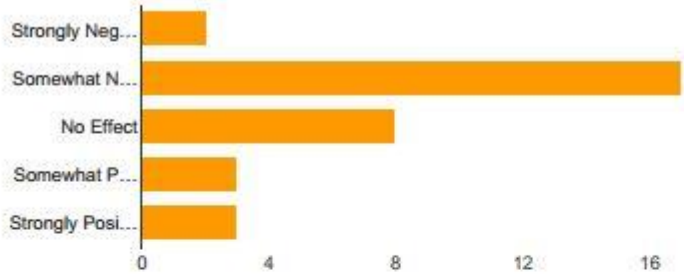
Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	19	57.6%
Somewhat Positive	4	12.1%
Strongly Positive	4	12.1%

↑ Degree of Cohesion [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]

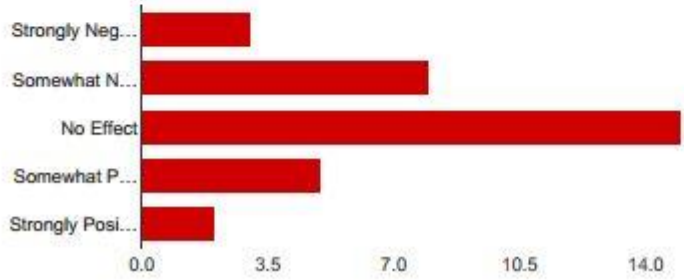


Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	19	57.6%
Somewhat Positive	5	15.2%
Strongly Positive	2	6.1%

↑ Num of Attributes [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]

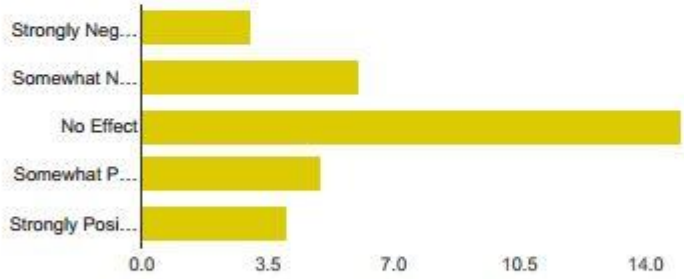


↑ Number of Methods [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



Strongly Negative	**3**	9.1%
Somewhat Negative	**8**	24.2%
No Effect	**15**	45.5%
Somewhat Positive	**5**	15.2%
Strongly Positive	**2**	6.1%

↑ Number of Intents [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



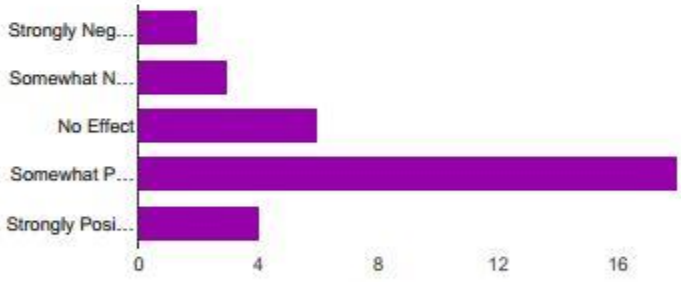
Strongly Negative	**3**	9.1%
Somewhat Negative	**6**	18.2%
No Effect	**15**	45.5%

Somewhat Positive	5	15.2%
Strongly Positive	4	12.1%

↑ Cyclomatic Complexity [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]

Strongly Negative	17	51.5%
Somewhat Negative	1	3%
No Effect	6	18.2%
Somewhat Positive	3	9.1%
Strongly Positive	6	18.2%

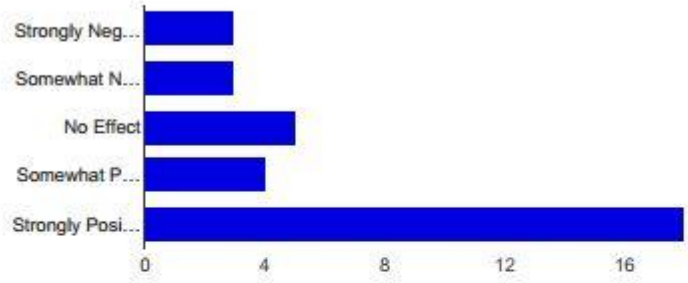
↑ Num of <uses-permission> [6- Select the degree of effects for each factors on the PERFORMANCE EFFICIENCY of Mobile Application.]



Strongly Negative	2	6.1%
Somewhat Negative	3	9.1%
No Effect	6	18.2%
Somewhat Positive	18	54.5%
Strongly Positive	4	12.1%

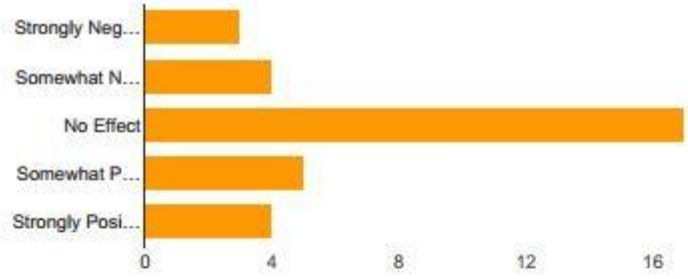
What other factors would you offer that affects PERFORMANCE EFFICIENCY?

↑ Num of Classes [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



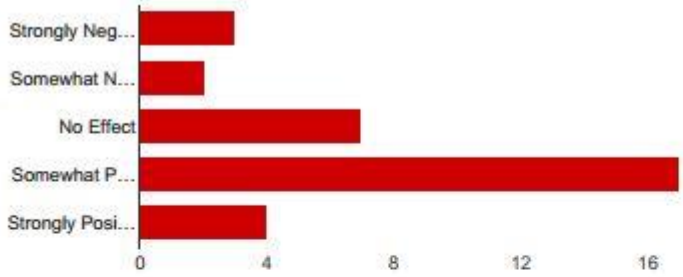
Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	4	12.1%
Strongly Positive	18	54.5%

↑ Num of Activities [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]

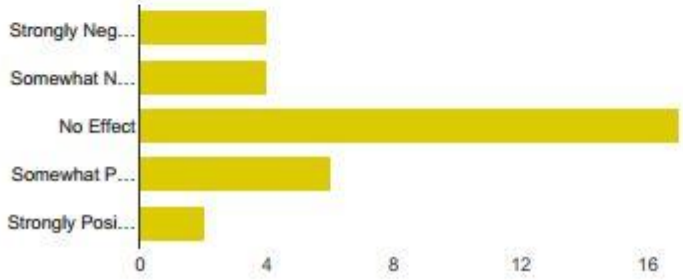


Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	17	51.5%
Somewhat Positive	5	15.2%
Strongly Positive	4	12.1%

↑ Num of Services [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]

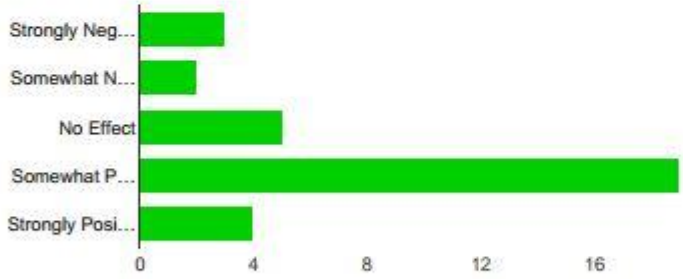


↑ Num of BroadcastReceivers [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	17	51.5%
Somewhat Positive	6	18.2%
Strongly Positive	2	6.1%

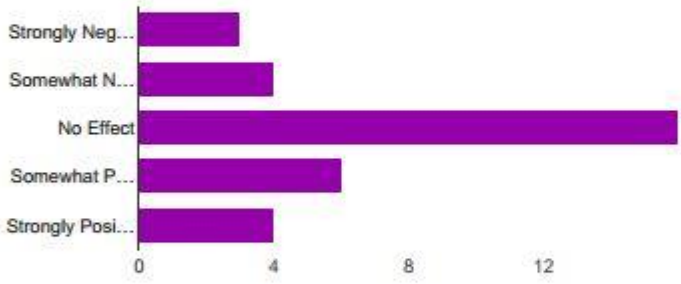
↑ Num of ContentProviders [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	2	6.1%

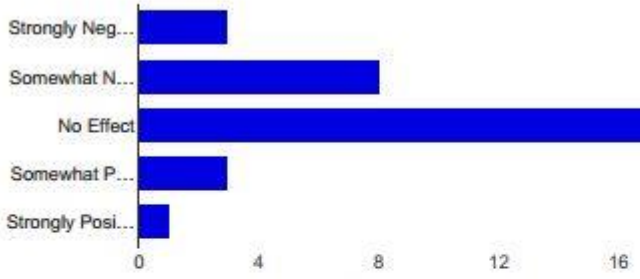
No Effect	5	15.2%
Somewhat Positive	19	57.6%
Strongly Positive	4	12.1%

↑ Number of Dialogs [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



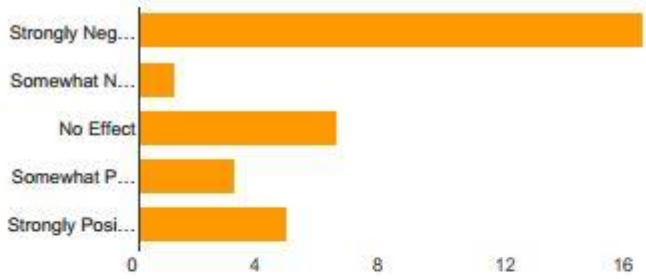
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	16	48.5%
Somewhat Positive	6	18.2%
Strongly Positive	4	12.1%

↑ Number of Threads [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



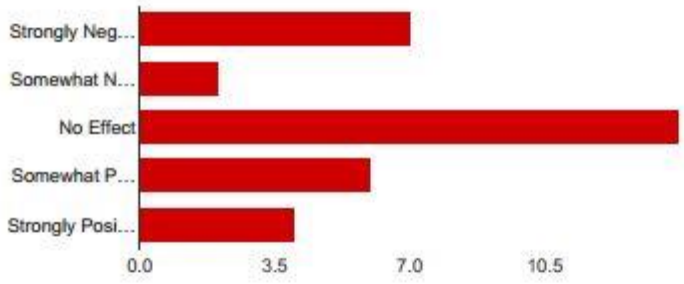
Strongly Negative	3	9.1%
Somewhat Negative	8	24.2%
No Effect	18	54.5%
Somewhat Positive	3	9.1%
Strongly Positive	1	3%

↑ android:minSdkVersion [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



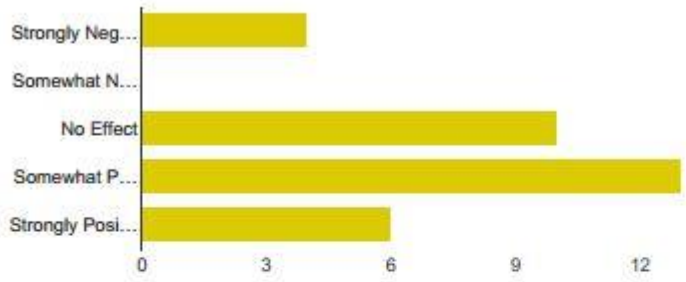
Strongly Negative	17	51.5%
Somewhat Negative	1	3%
No Effect	7	21.2%
Somewhat Positive	3	9.1%
Strongly Positive	5	15.2%

↑ android:targetSdkVersion [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]

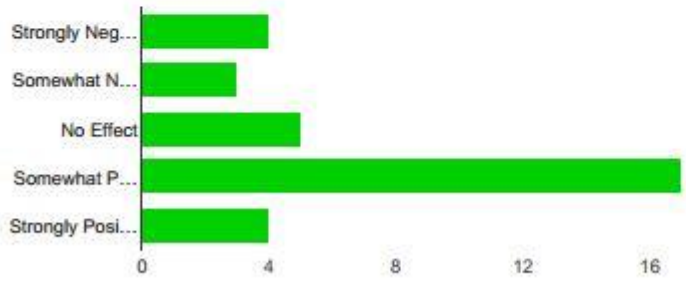


Strongly Negative	7	21.2%
Somewhat Negative	2	6.1%
No Effect	14	42.4%
Somewhat Positive	6	18.2%
Strongly Positive	4	12.1%

↑ Num of Tables in Database [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]

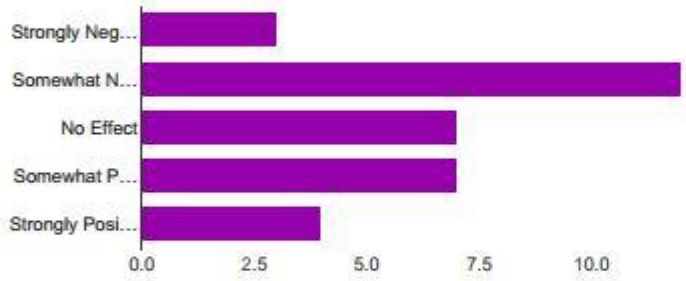


↑ Depth of Inheritance Tree [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



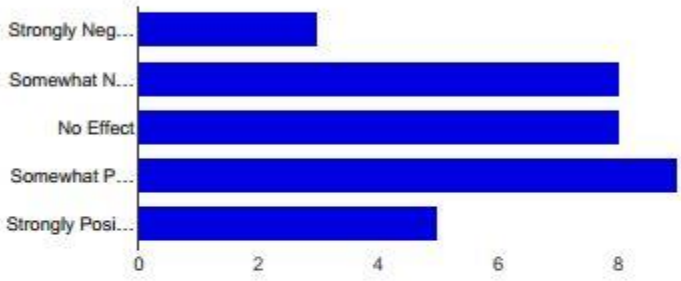
Strongly Negative	**4**	12.1%
Somewhat Negative	**3**	9.1%
No Effect	**5**	15.2%
Somewhat Positive	**17**	51.5%
Strongly Positive	**4**	12.1%

↑ Degree of Coupling [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



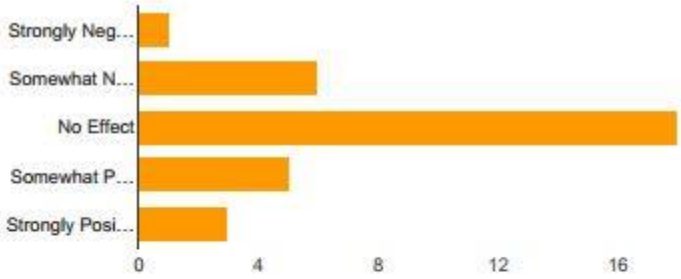
Strongly Negative	**3**	9.1%
Somewhat Negative	**12**	36.4%
No Effect	**7**	21.2%
Somewhat Positive	**7**	21.2%
Strongly Positive	**4**	12.1%

↑ Degree of Cohesion [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



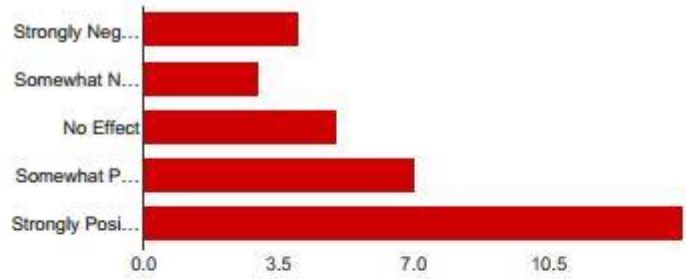
Strongly Negative	3	9.1%
Somewhat Negative	8	24.2%
No Effect	8	24.2%
Somewhat Positive	9	27.3%
Strongly Positive	5	15.2%

↑ Num of Attributes [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



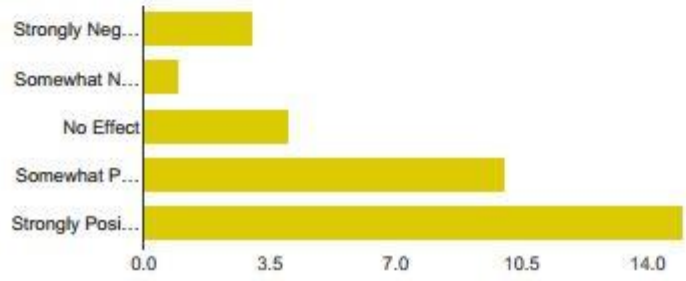
Strongly Negative	1	3%
Somewhat Negative	6	18.2%
No Effect	18	54.5%
Somewhat Positive	5	15.2%
Strongly Positive	3	9.1%

↑ Number of Methods [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



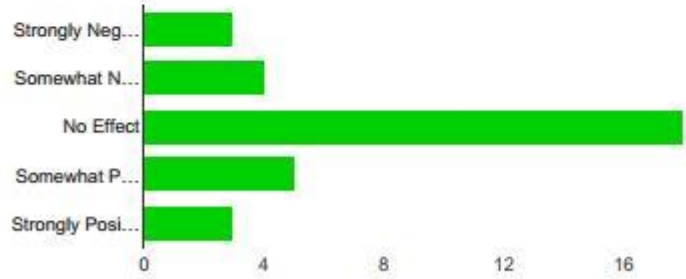
Strongly Negative	4	12.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	7	21.2%
Strongly Positive	14	42.4%

↑ Number of Intents [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



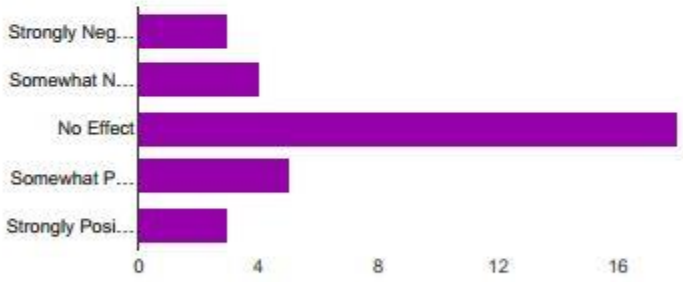
Strongly Negative	3	9.1%
Somewhat Negative	1	3%
No Effect	4	12.1%
Somewhat Positive	10	30.3%
Strongly Positive	15	45.5%

↑ Cyclomatic Complexity [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	18	54.5%
Somewhat Positive	5	15.2%
Strongly Positive	3	9.1%

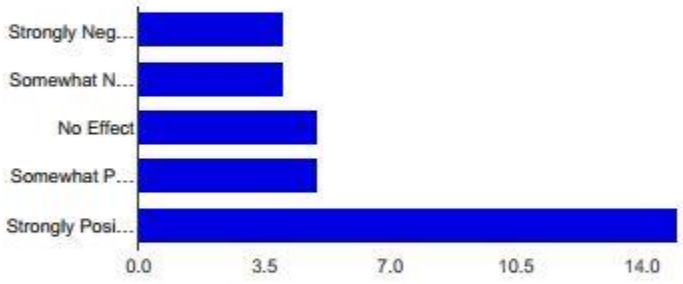
↑ Num of <uses-permission> [7- Select the degree of effects for each factors on the PORTABILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	18	54.5%
Somewhat Positive	5	15.2%
Strongly Positive	3	9.1%

What other factors would you offer that affects PORTABILITY?

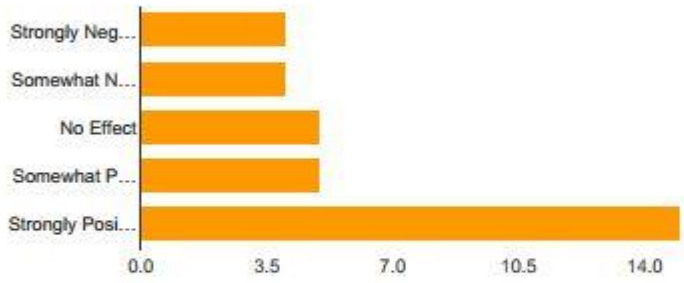
↑ Num of Classes [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	5	15.2%
Somewhat Positive	5	15.2%

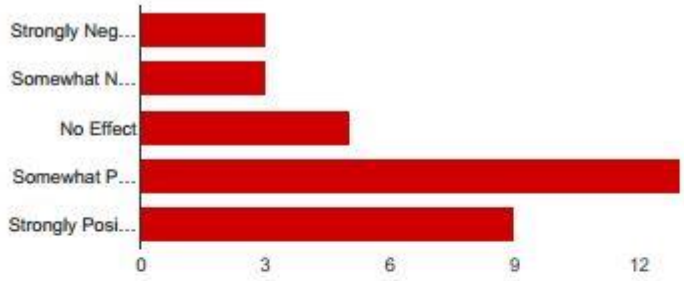
Strongly Positive	15	45.5%

↑ Num of Activities [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



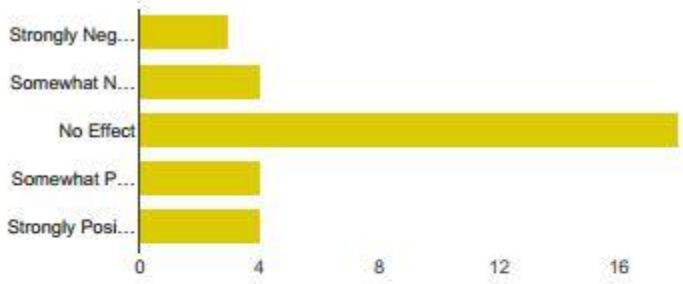
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	5	15.2%
Somewhat Positive	5	15.2%
Strongly Positive	15	45.5%

↑ Num of Services [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



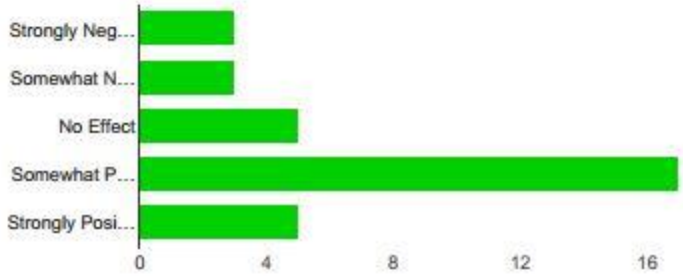
Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	13	39.4%
Strongly Positive	9	27.3%

↑ Num of BroadcastReceivers [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	18	54.5%
Somewhat Positive	4	12.1%
Strongly Positive	4	12.1%

↑ Num of ContentProviders [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]

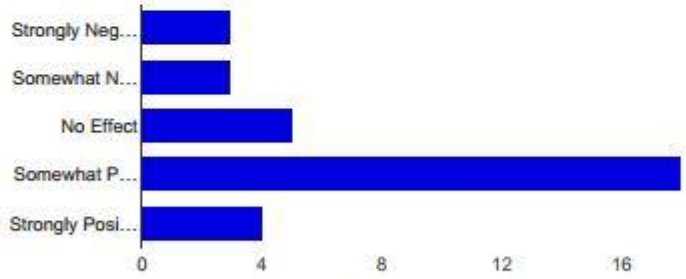


Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	17	51.5%
Strongly Positive	5	15.2%

↑ Number of Dialogs [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]

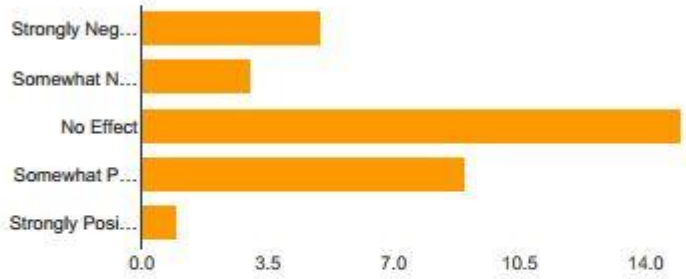
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	18	54.5%
Somewhat Positive	4	12.1%
Strongly Positive	3	9.1%

↑ Number of Threads [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



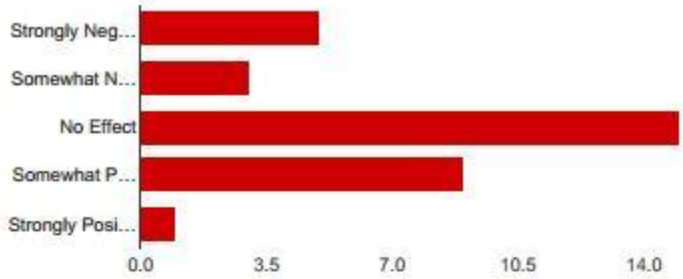
Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	18	54.5%
Strongly Positive	4	12.1%

↑ android:minSdkVersion [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



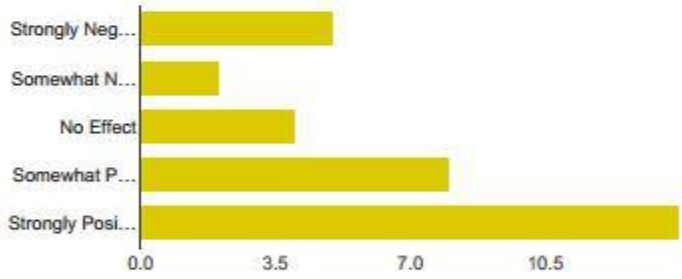
Strongly Negative	5	15.2%
Somewhat Negative	3	9.1%
No Effect	15	45.5%
Somewhat Positive	9	27.3%
Strongly Positive	1	3%

↑ android:targetSdkVersion [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



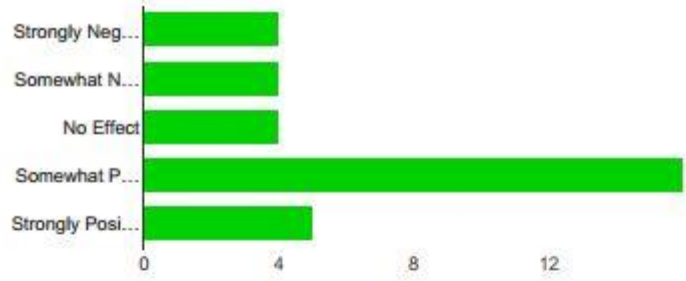
Strongly Negative	5	15.2%
Somewhat Negative	3	9.1%
No Effect	15	45.5%
Somewhat Positive	9	27.3%
Strongly Positive	1	3%

↑ Num of Tables in Database [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]

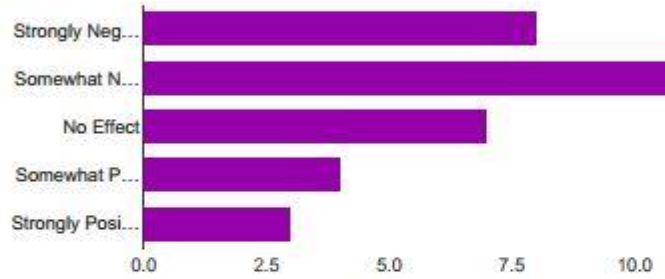


Strongly Negative	5	15.2%
Somewhat Negative	2	6.1%
No Effect	4	12.1%
Somewhat Positive	8	24.2%
Strongly Positive	14	42.4%

↑ Depth of Inheritance Tree [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]

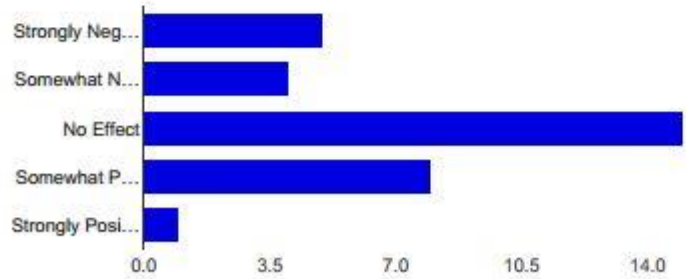


↑ Degree of Coupling [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



Strongly Negative	8	24.2%
Somewhat Negative	11	33.3%
No Effect	7	21.2%
Somewhat Positive	4	12.1%
Strongly Positive	3	9.1%

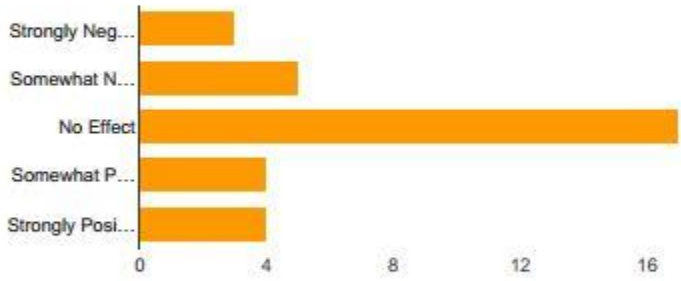
↑ Degree of Cohesion [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



Strongly Negative	5	15.2%
Somewhat Negative	4	12.1%

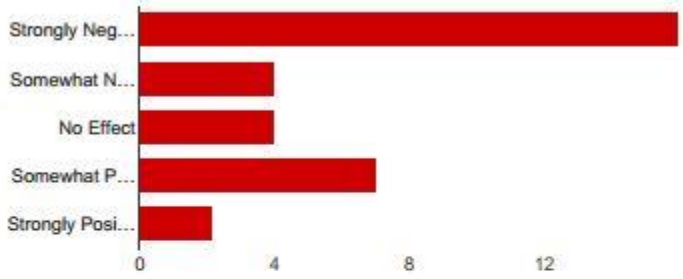
No Effect	15	45.5%
Somewhat Positive	8	24.2%
Strongly Positive	1	3%

↑ Num of Attributes [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



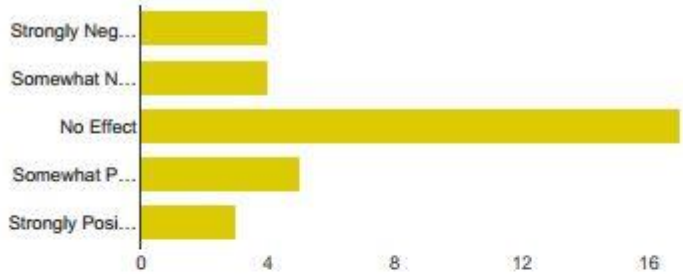
Strongly Negative	3	9.1%
Somewhat Negative	5	15.2%
No Effect	17	51.5%
Somewhat Positive	4	12.1%
Strongly Positive	4	12.1%

↑ Number of Methods [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



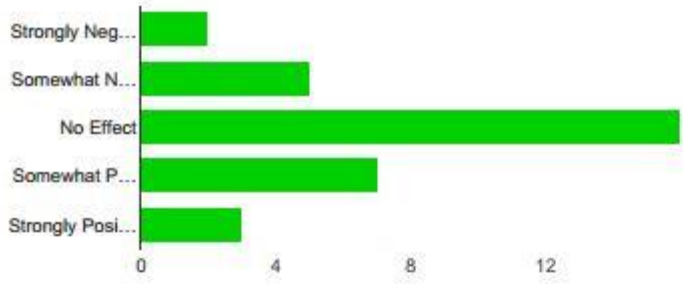
Strongly Negative	16	48.5%
Somewhat Negative	4	12.1%
No Effect	4	12.1%
Somewhat Positive	7	21.2%
Strongly Positive	2	6.1%

↑ Number of Intents [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



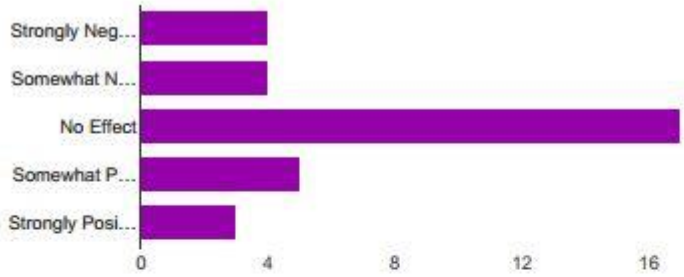
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	17	51.5%
Somewhat Positive	5	15.2%
Strongly Positive	3	9.1%

↑ Cyclomatic Complexity [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



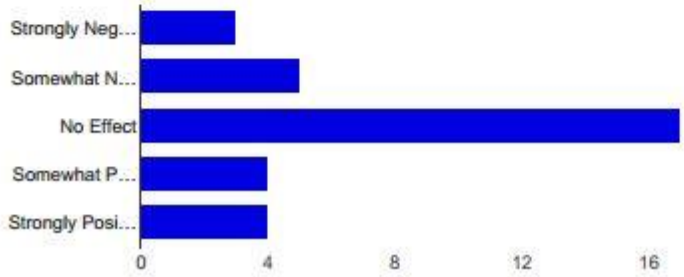
Strongly Negative	2	6.1%
Somewhat Negative	5	15.2%
No Effect	16	48.5%
Somewhat Positive	7	21.2%
Strongly Positive	3	9.1%

↑ Num of <uses-permission> [8- Select the degree of effects for each factors on the MAINTAINABILITY of Mobile Application.]



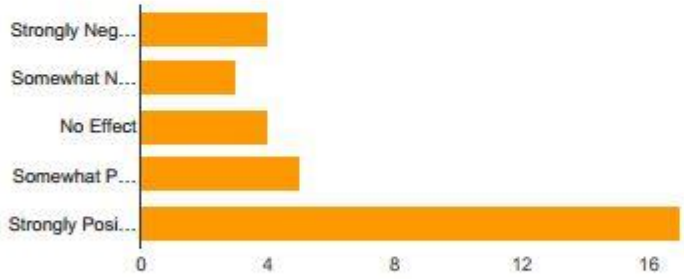
What other factors would you offer that affects MAINTAINABILITY?

↑ Num of Classes [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



Strongly Negative	**3**	9.1%
Somewhat Negative	**5**	15.2%
No Effect	**17**	51.5%
Somewhat Positive	**4**	12.1%
Strongly Positive	**4**	12.1%

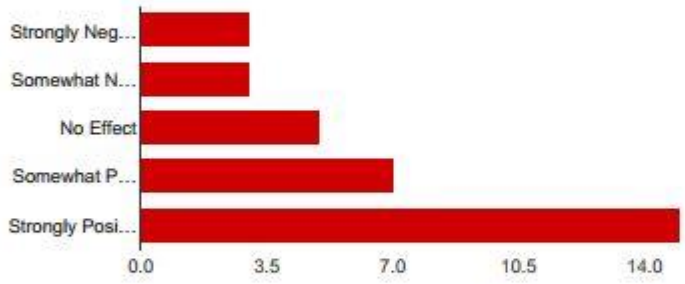
↑ Num of Activities [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



Strongly Negative	**4**	12.1%
Somewhat Negative	**3**	9.1%

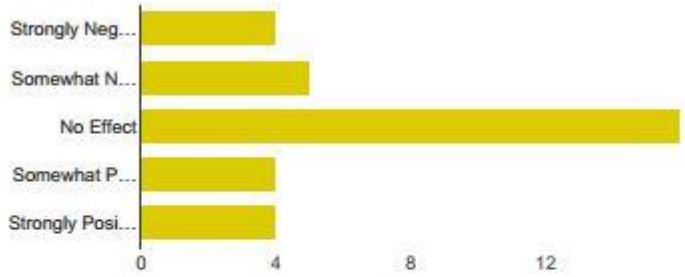
No Effect	4	12.1%
Somewhat Positive	5	15.2%
Strongly Positive	17	51.5%

↑ Num of Services [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



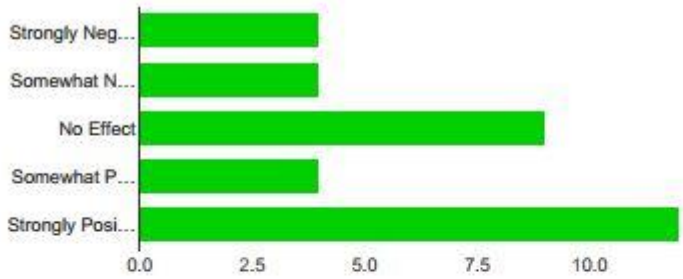
Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	5	15.2%
Somewhat Positive	7	21.2%
Strongly Positive	15	45.5%

↑ Num of BroadcastReceivers [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



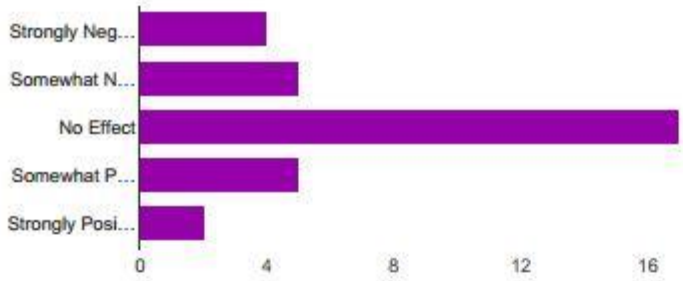
Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	16	48.5%
Somewhat Positive	4	12.1%
Strongly Positive	4	12.1%

↑ Num of ContentProviders [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



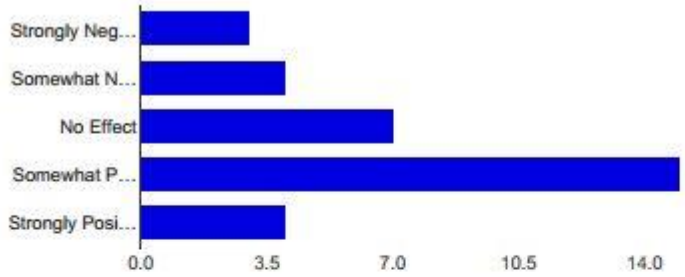
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	9	27.3%
Somewhat Positive	4	12.1%
Strongly Positive	12	36.4%

↑ Number of Dialogs [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]

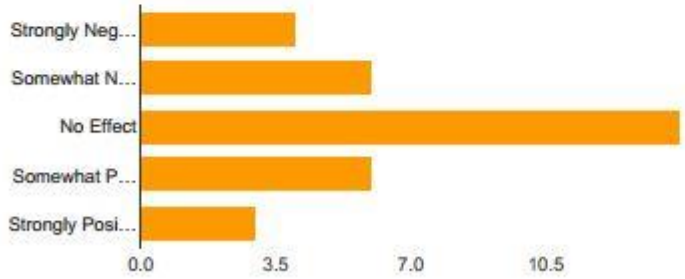


Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	17	51.5%
Somewhat Positive	5	15.2%
Strongly Positive	2	6.1%

↑ Number of Threads [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]

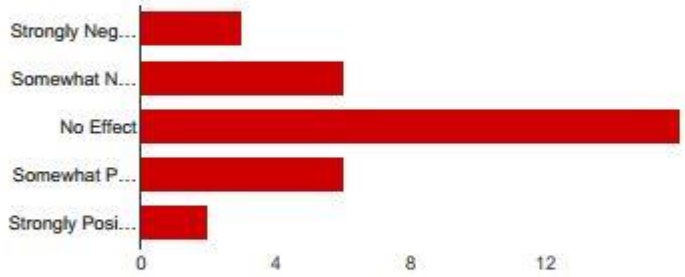


↑ android:minSdkVersion [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



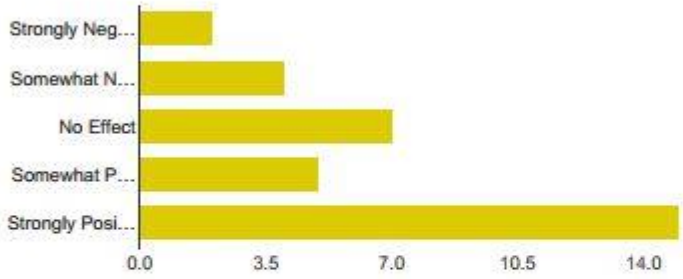
Strongly Negative	4	12.1%
Somewhat Negative	6	18.2%
No Effect	14	42.4%
Somewhat Positive	6	18.2%
Strongly Positive	3	9.1%

↑ android:targetSdkVersion [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



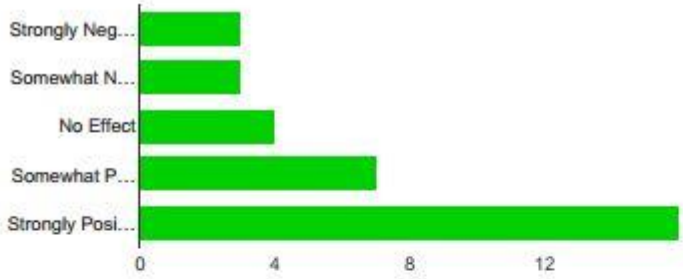
Strongly Negative	3	9.1%
Somewhat Negative	6	18.2%
No Effect	16	48.5%
Somewhat Positive	6	18.2%
Strongly Positive	2	6.1%

↑ Num of Tables in Database [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



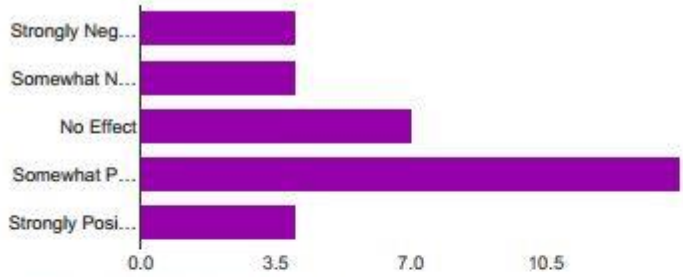
Strongly Negative	**2**	6.1%
Somewhat Negative	**4**	12.1%
No Effect	**7**	21.2%
Somewhat Positive	**5**	15.2%
Strongly Positive	**15**	45.5%

↑ Depth of Inheritance Tree [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



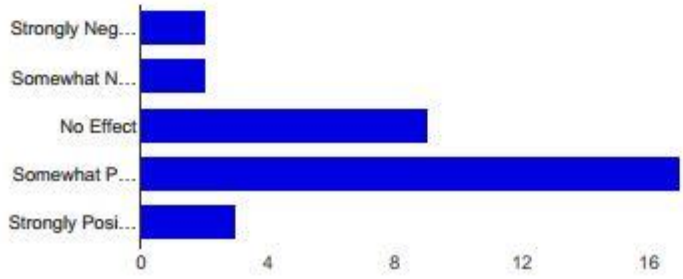
Strongly Negative	**3**	9.1%
Somewhat Negative	**3**	9.1%
No Effect	**4**	12.1%
Somewhat Positive	**7**	21.2%
Strongly Positive	**16**	48.5%

↑ Degree of Coupling [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



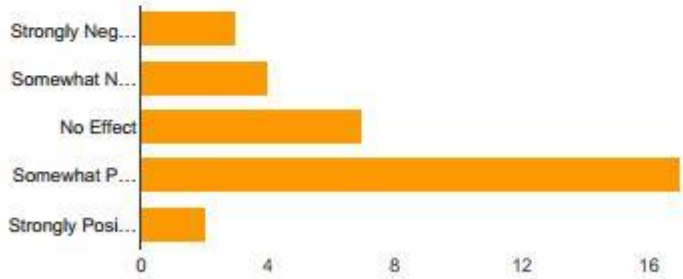
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	7	21.2%
Somewhat Positive	14	42.4%
Strongly Positive	4	12.1%

↑ Degree of Cohesion [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



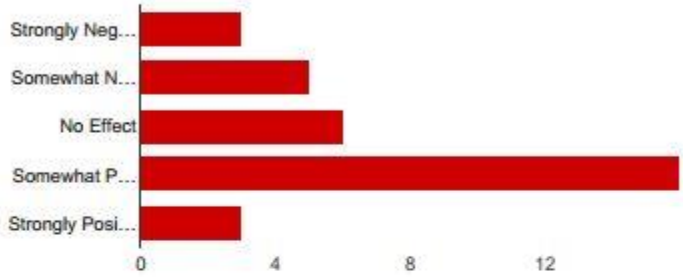
Strongly Negative	2	6.1%
Somewhat Negative	2	6.1%
No Effect	9	27.3%
Somewhat Positive	17	51.5%
Strongly Positive	3	9.1%

↑ Num of Attributes [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



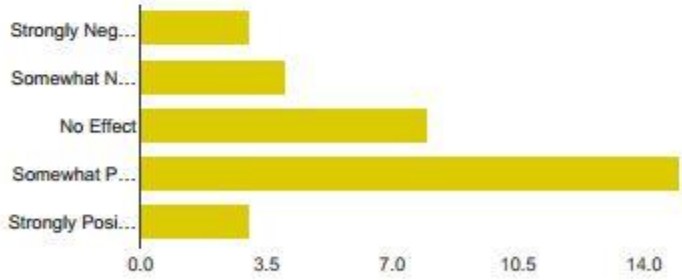
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	7	21.2%
Somewhat Positive	17	51.5%
Strongly Positive	2	6.1%

↑ Number of Methods [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



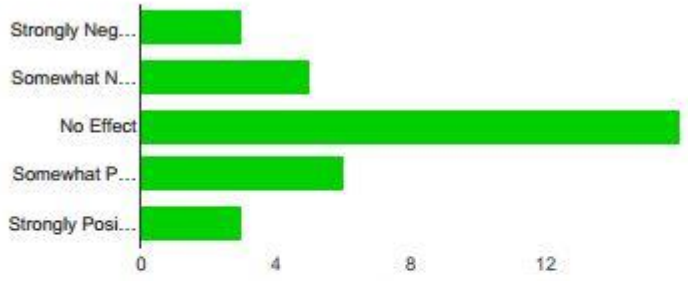
Strongly Negative	3	9.1%
Somewhat Negative	5	15.2%
No Effect	6	18.2%
Somewhat Positive	16	48.5%
Strongly Positive	3	9.1%

↑ Number of Intents [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



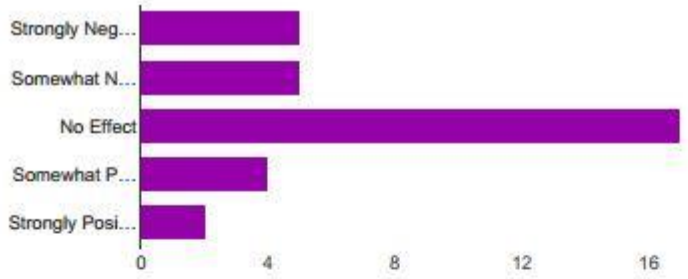
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	8	24.2%
Somewhat Positive	15	45.5%
Strongly Positive	3	9.1%

↑ Cyclomatic Complexity [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	5	15.2%
No Effect	16	48.5%
Somewhat Positive	6	18.2%
Strongly Positive	3	9.1%

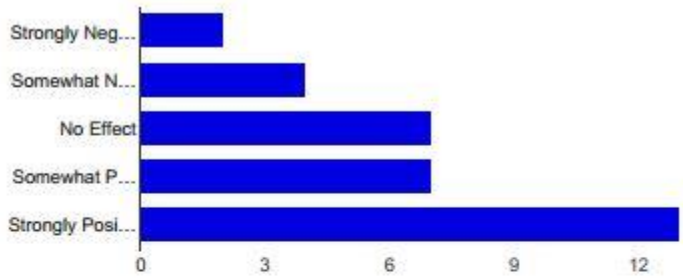
↑ Num of <uses-permission> [9- Select the degree of effects for each factors on the DATA INTEGRITY of Mobile Application.]



Strongly Negative	5	15.2%
Somewhat Negative	5	15.2%
No Effect	17	51.5%
Somewhat Positive	4	12.1%
Strongly Positive	2	6.1%

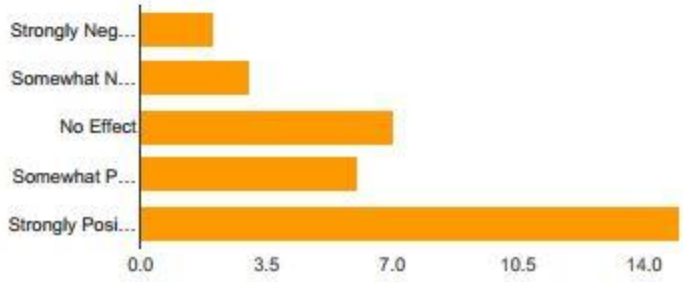
What other factors would you offer that affects DATA INTEGRITY?

↑ Num of Classes [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



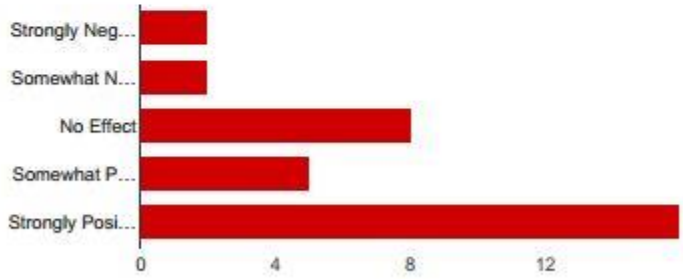
Strongly Negative	2	6.1%
Somewhat Negative	4	12.1%
No Effect	7	21.2%
Somewhat Positive	7	21.2%
Strongly Positive	13	39.4%

↑ Num of Activities [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



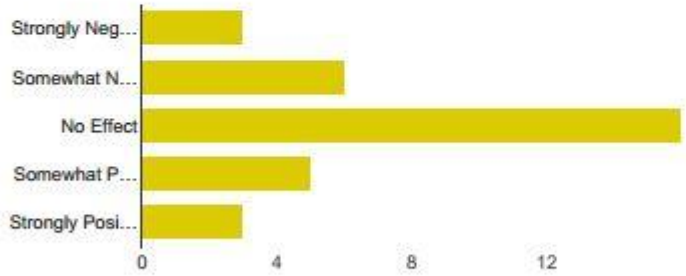
Strongly Negative	2	6.1%
Somewhat Negative	3	9.1%
No Effect	7	21.2%
Somewhat Positive	6	18.2%
Strongly Positive	15	45.5%

↑ Num of Services [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



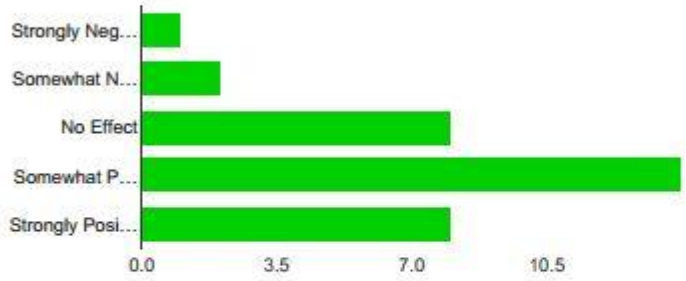
Strongly Negative	2	6.1%
Somewhat Negative	2	6.1%
No Effect	8	24.2%
Somewhat Positive	5	15.2%
Strongly Positive	16	48.5%

↑ Num of BroadcastReceivers [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



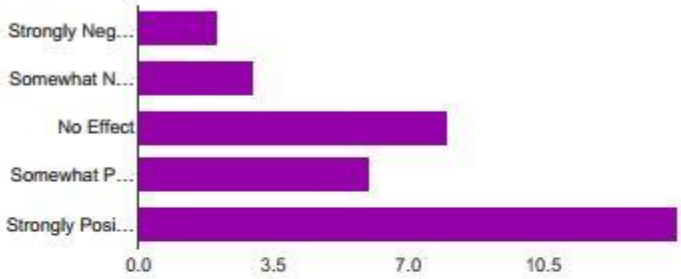
Strongly Negative	3	9.1%
Somewhat Negative	6	18.2%
No Effect	16	48.5%
Somewhat Positive	5	15.2%
Strongly Positive	3	9.1%

↑ Num of ContentProviders [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



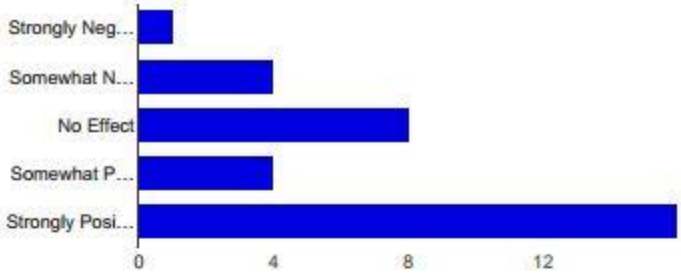
Strongly Negative	1	3%
Somewhat Negative	2	6.1%
No Effect	8	24.2%
Somewhat Positive	14	42.4%
Strongly Positive	8	24.2%

↑ Number of Dialogs [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



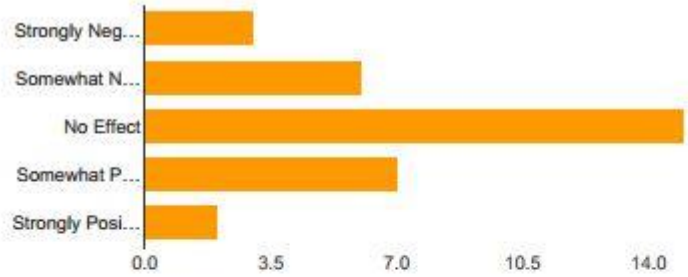
Strongly Negative	**2**	6.1%
Somewhat Negative	**3**	9.1%
No Effect	**8**	24.2%
Somewhat Positive	**6**	18.2%
Strongly Positive	**14**	42.4%

↑ Number of Threads [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

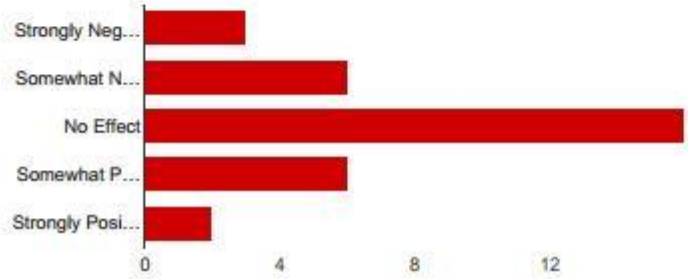


Strongly Negative	**1**	3%
Somewhat Negative	**4**	12.1%
No Effect	**8**	24.2%
Somewhat Positive	**4**	12.1%
Strongly Positive	**16**	48.5%

↑ android:minSdkVersion [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

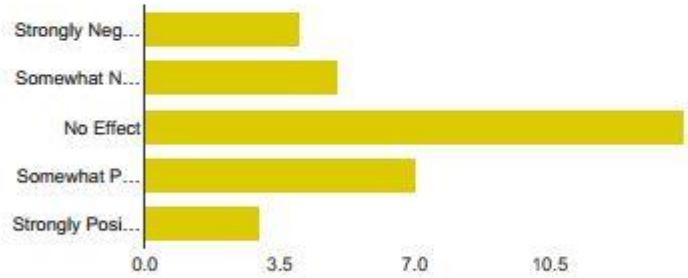


↑ android:targetSdkVersion [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



Strongly Negative	**3**	9.1%
Somewhat Negative	**6**	18.2%
No Effect	**16**	48.5%
Somewhat Positive	**6**	18.2%
Strongly Positive	**2**	6.1%

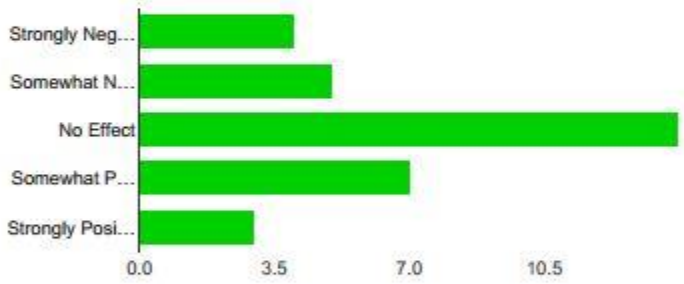
↑ Num of Tables in Database [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



Strongly Negative	**4**	12.1%
Somewhat Negative	**5**	15.2%

No Effect	14	42.4%
Somewhat Positive	7	21.2%
Strongly Positive	3	9.1%

↑ Depth of Inheritance Tree [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

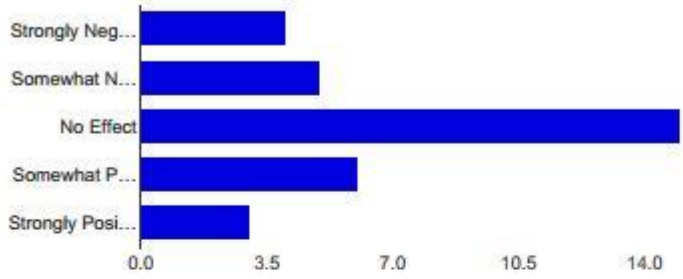


Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	14	42.4%
Somewhat Positive	7	21.2%
Strongly Positive	3	9.1%

↑ Degree of Coupling [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

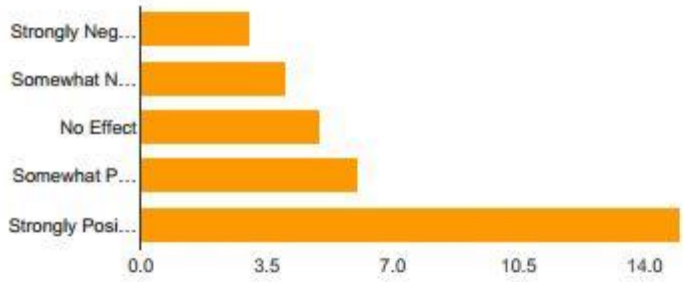
Strongly Negative	4	12.1%
Somewhat Negative	4	12.1%
No Effect	16	48.5%
Somewhat Positive	6	18.2%
Strongly Positive	3	9.1%

↑ Degree of Cohesion [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



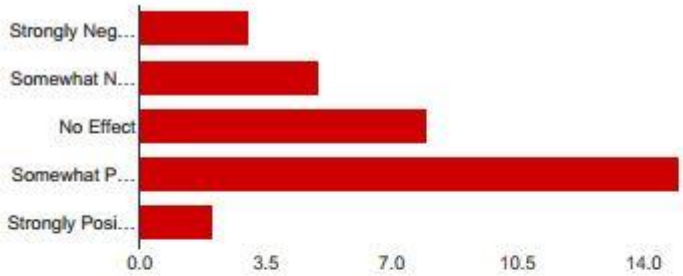
Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	15	45.5%
Somewhat Positive	6	18.2%
Strongly Positive	3	9.1%

↑ Num of Attributes [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

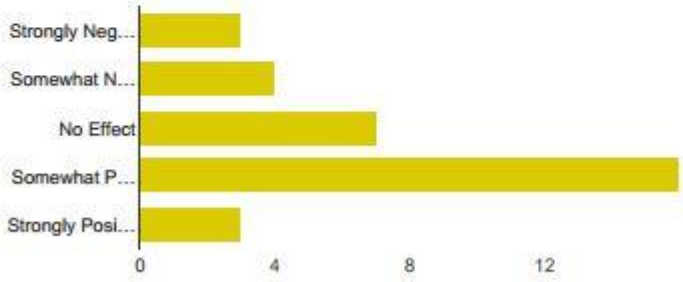


Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	5	15.2%
Somewhat Positive	6	18.2%
Strongly Positive	15	45.5%

↑ Number of Methods [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

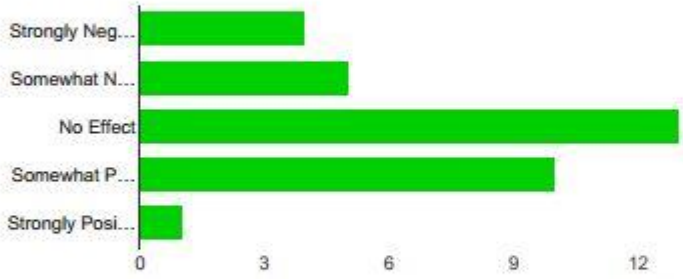


↑ Number of Intents [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	7	21.2%
Somewhat Positive	16	48.5%
Strongly Positive	3	9.1%

↑ Cyclomatic Complexity [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]



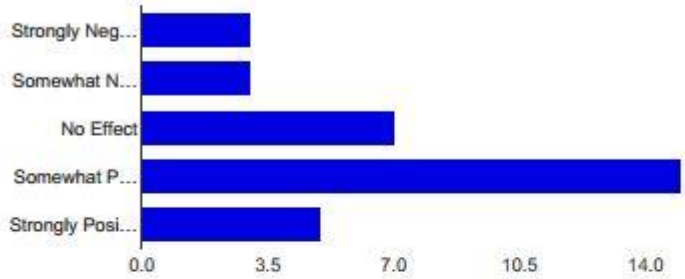
Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	13	39.4%
Somewhat Positive	10	30.3%
Strongly Positive	1	3%

↑ Num of <uses-permission> [10- Select the degree of effects for each factors on the USABILITY of Mobile Application.]

Strongly Negative	**14**	42.4%
Somewhat Negative	**4**	12.1%
No Effect	**6**	18.2%
Somewhat Positive	**8**	24.2%
Strongly Positive	**1**	3%

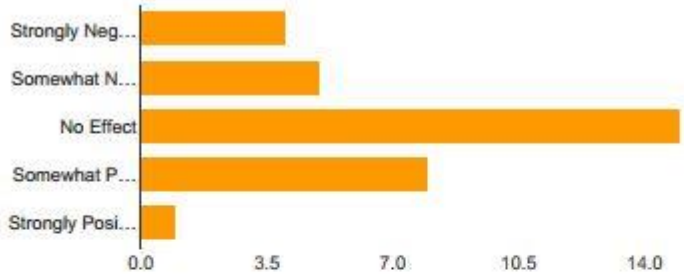
What other factors would you offer that affects USABILITY?

↑ Num of Classes [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



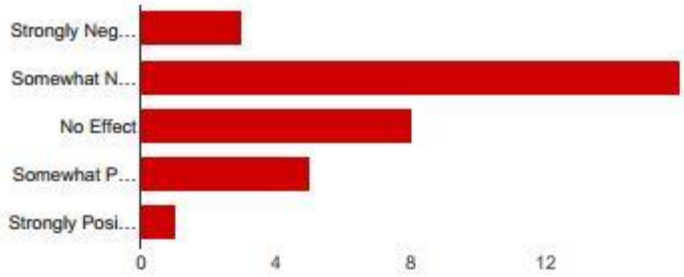
Strongly Negative	**3**	9.1%
Somewhat Negative	**3**	9.1%
No Effect	**7**	21.2%
Somewhat Positive	**15**	45.5%
Strongly Positive	**5**	15.2%

↑ Num of Activities [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



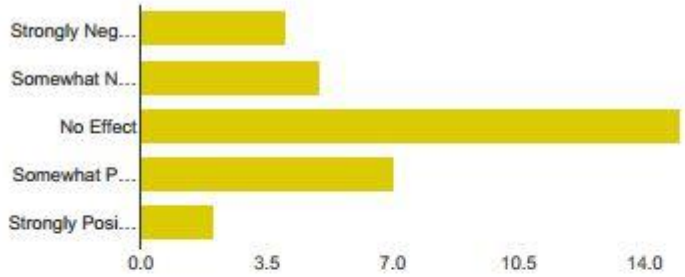
Strongly Negative	4	12.1%
Somewhat Negative	5	15.2%
No Effect	15	45.5%
Somewhat Positive	8	24.2%
Strongly Positive	1	3%

↑ Num of Services [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]

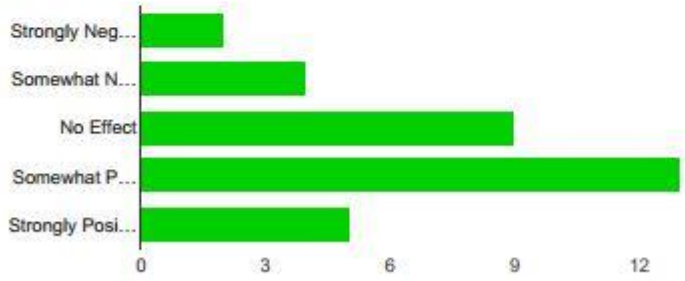


Strongly Negative	3	9.1%
Somewhat Negative	16	48.5%
No Effect	8	24.2%
Somewhat Positive	5	15.2%
Strongly Positive	1	3%

↑ Num of BroadcastReceivers [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]

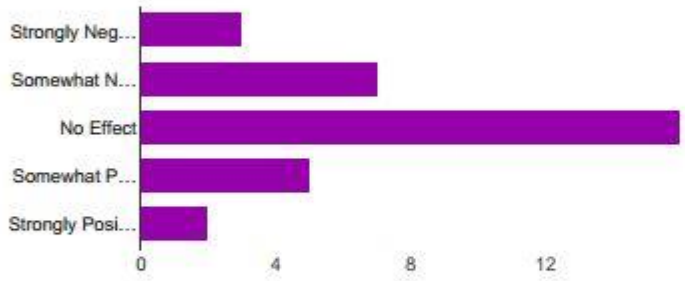


↑ Num of ContentProviders [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



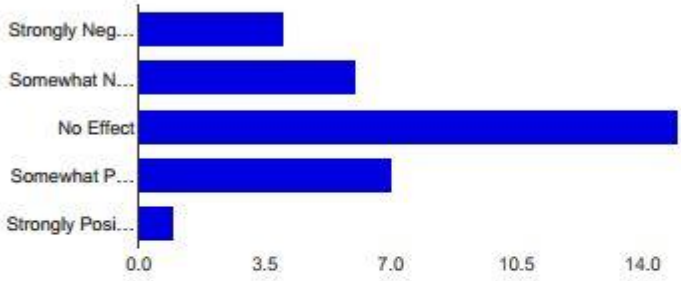
Strongly Negative	**2**	6.1%
Somewhat Negative	**4**	12.1%
No Effect	**9**	27.3%
Somewhat Positive	**13**	39.4%
Strongly Positive	**5**	15.2%

↑ Number of Dialogs [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



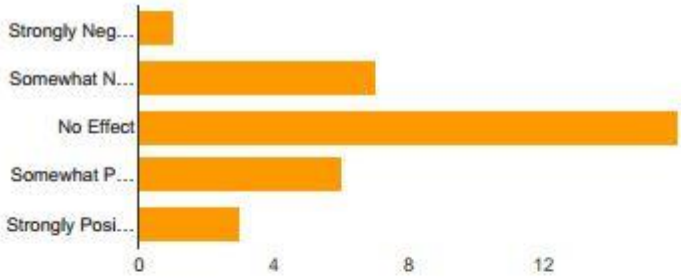
Strongly Negative	**3**	9.1%
Somewhat Negative	**7**	21.2%
No Effect	**16**	48.5%
Somewhat Positive	**5**	15.2%
Strongly Positive	**2**	6.1%

↑ Number of Threads [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



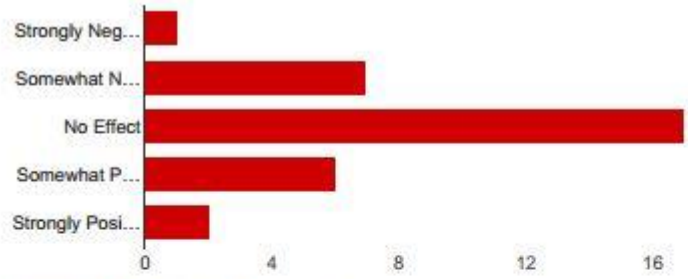
Strongly Negative	4	12.1%
Somewhat Negative	6	18.2%
No Effect	15	45.5%
Somewhat Positive	7	21.2%
Strongly Positive	1	3%

↑ android:minSdkVersion [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



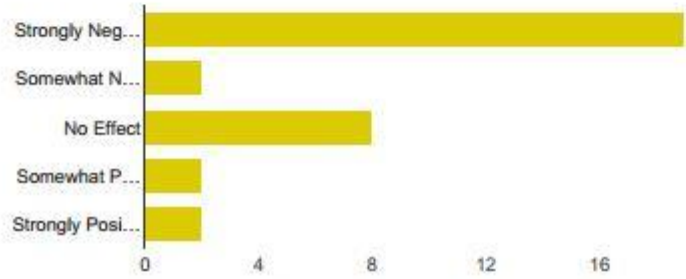
Strongly Negative	1	3%
Somewhat Negative	7	21.2%
No Effect	16	48.5%
Somewhat Positive	6	18.2%
Strongly Positive	3	9.1%

↑ android:targetSdkVersion [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



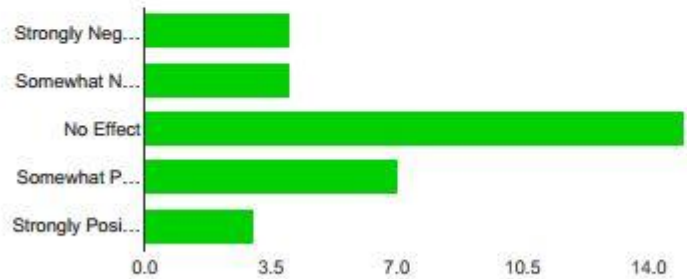
Strongly Negative	1	3%
Somewhat Negative	7	21.2%
No Effect	17	51.5%
Somewhat Positive	6	18.2%
Strongly Positive	2	6.1%

↑ Num of Tables in Database [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



Strongly Negative	19	57.6%
Somewhat Negative	2	6.1%
No Effect	8	24.2%
Somewhat Positive	2	6.1%
Strongly Positive	2	6.1%

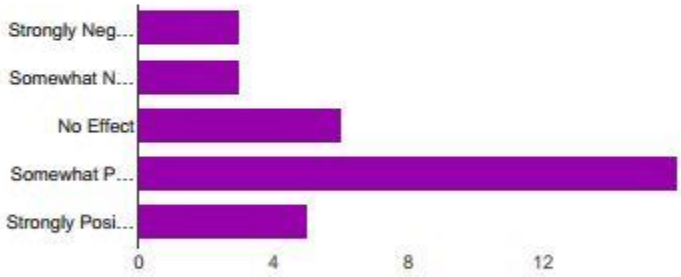
↑ Depth of Inheritance Tree [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



Strongly Negative	4	12.1%

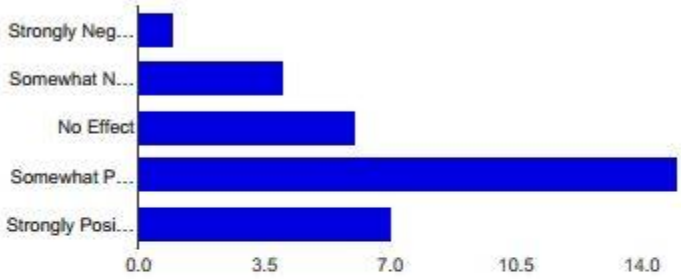
Somewhat Negative	4	12.1%
No Effect	15	45.5%
Somewhat Positive	7	21.2%
Strongly Positive	3	9.1%

↑ Degree of Coupling [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	6	18.2%
Somewhat Positive	16	48.5%
Strongly Positive	5	15.2%

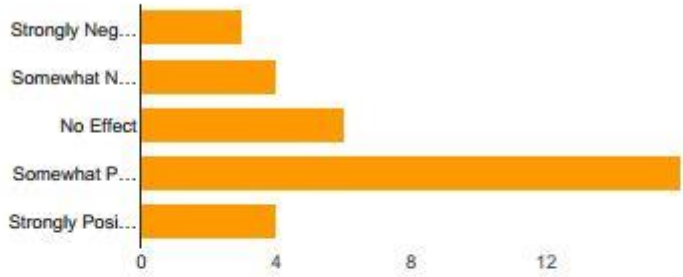
↑ Degree of Cohesion [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



Strongly Negative	1	3%
Somewhat Negative	4	12.1%
No Effect	6	18.2%
Somewhat Positive	15	45.5%
Strongly Positive	7	21.2%

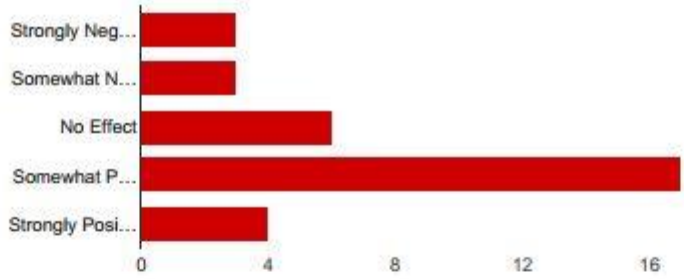
↑ Num of Attributes [11- Select the degree of effects for each factors on

the SECURITY of Mobile Application.]



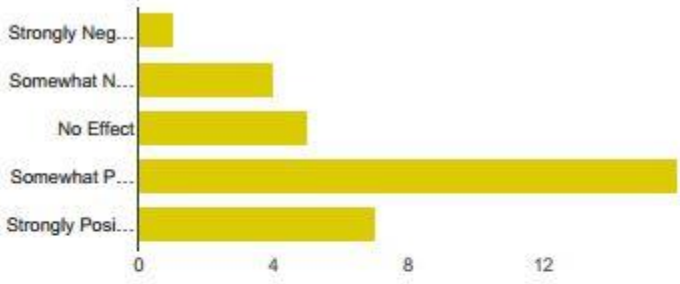
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	6	18.2%
Somewhat Positive	16	48.5%
Strongly Positive	4	12.1%

↑ Number of Methods [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]

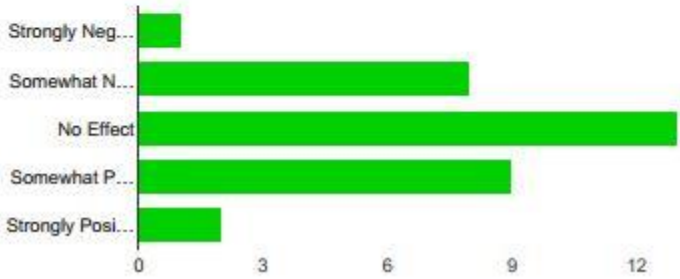


Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	6	18.2%
Somewhat Positive	17	51.5%
Strongly Positive	4	12.1%

↑ Number of Intents [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]

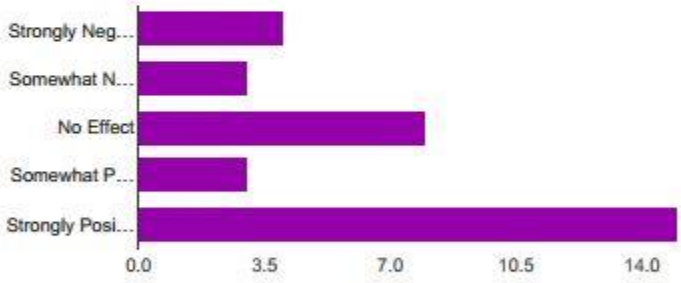


↑ Cyclomatic Complexity [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]



Strongly Negative	1	3%
Somewhat Negative	8	24.2%
No Effect	13	39.4%
Somewhat Positive	9	27.3%
Strongly Positive	2	6.1%

↑ Num of <uses-permission> [11- Select the degree of effects for each factors on the SECURITY of Mobile Application.]

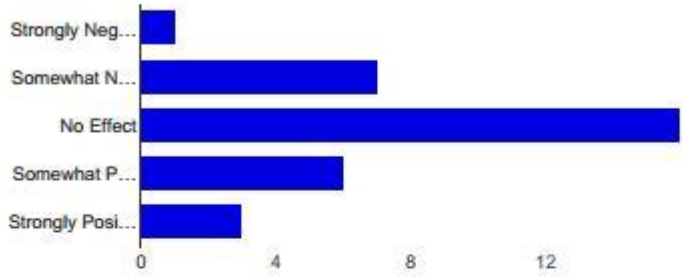


Strongly Negative	4	12.1%
Somewhat Negative	3	9.1%
No Effect	8	24.2%

Somewhat Positive	3	9.1%
Strongly Positive	15	45.5%

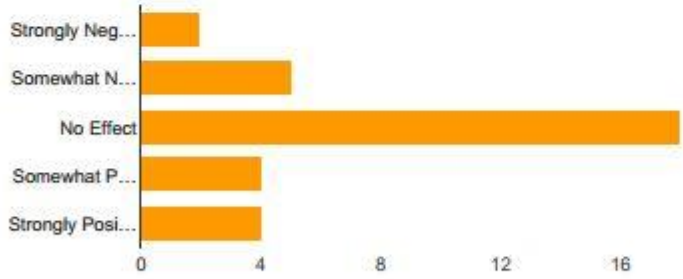
What other factors would you offer that affects SECURITY?

↑ Num of Classes [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



Strongly Negative	1	3%
Somewhat Negative	7	21.2%
No Effect	16	48.5%
Somewhat Positive	6	18.2%
Strongly Positive	3	9.1%

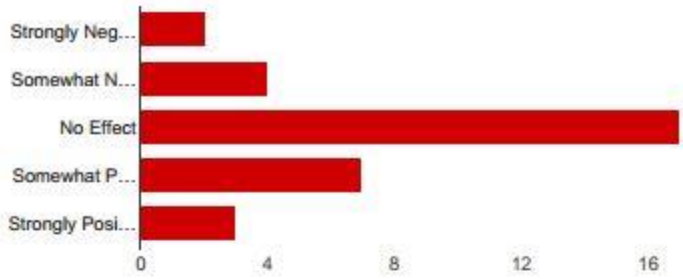
↑ Num of Activities [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



Strongly Negative	2	6.1%
Somewhat Negative	5	15.2%
No Effect	18	54.5%
Somewhat Positive	4	12.1%
Strongly Positive	4	12.1%

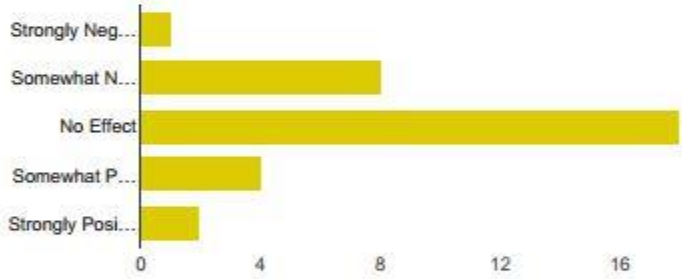
↑ Num of Services [12- Select the degree of effects for each factors on the

COMPATIBILITY of Mobile Application.]



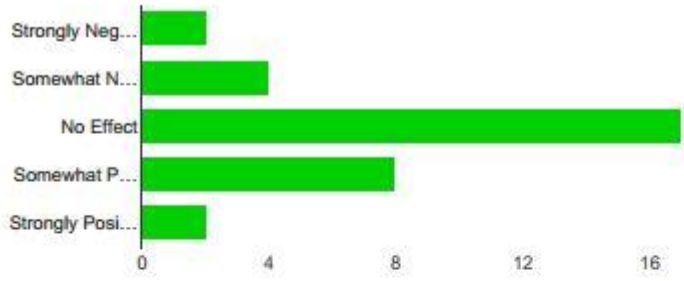
Strongly Negative	**2**	6.1%
Somewhat Negative	**4**	12.1%
No Effect	**17**	51.5%
Somewhat Positive	**7**	21.2%
Strongly Positive	**3**	9.1%

↑ Num of BroadcastReceivers [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]

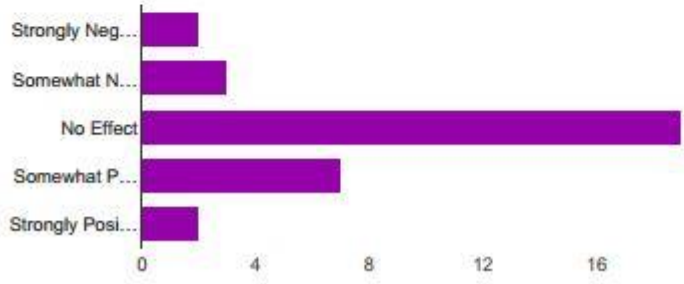


Strongly Negative	**1**	3%
Somewhat Negative	**8**	24.2%
No Effect	**18**	54.5%
Somewhat Positive	**4**	12.1%
Strongly Positive	**2**	6.1%

↑ Num of ContentProviders [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]

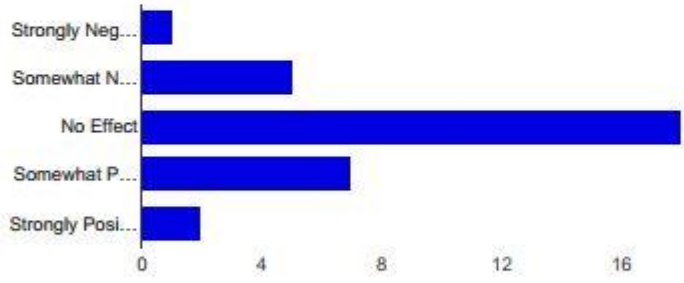


↑ Number of Dialogs [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



Strongly Negative	**2**	6.1%
Somewhat Negative	**3**	9.1%
No Effect	**19**	57.6%
Somewhat Positive	**7**	21.2%
Strongly Positive	**2**	6.1%

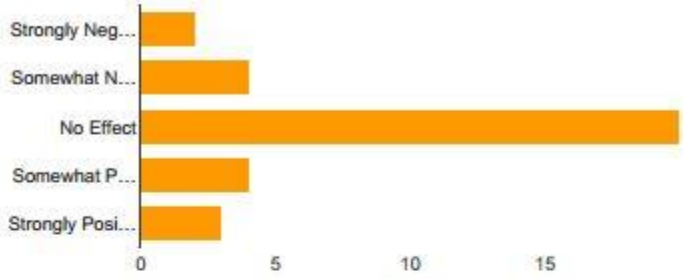
↑ Number of Threads [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



Strongly Negative	**1**	3%
Somewhat Negative	**5**	15.2%
No Effect	**18**	54.5%

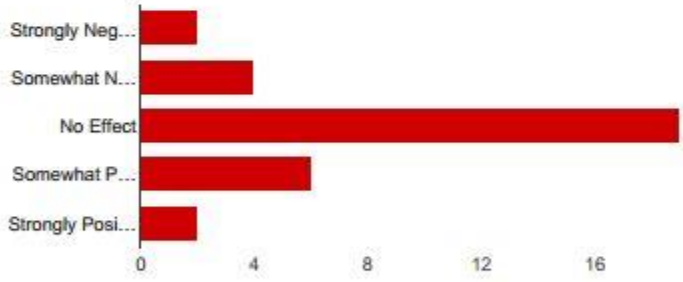
Somewhat Positive	7	21.2%
Strongly Positive	2	6.1%

↑ android:minSdkVersion [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



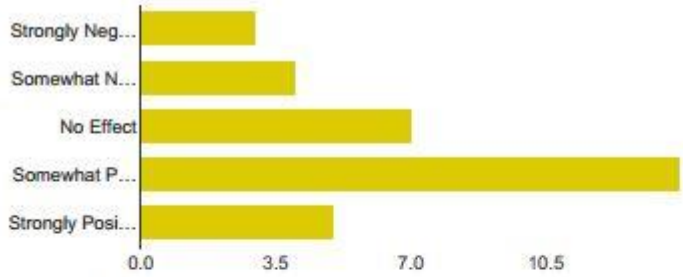
Strongly Negative	2	6.1%
Somewhat Negative	4	12.1%
No Effect	20	60.6%
Somewhat Positive	4	12.1%
Strongly Positive	3	9.1%

↑ android:targetSdkVersion [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



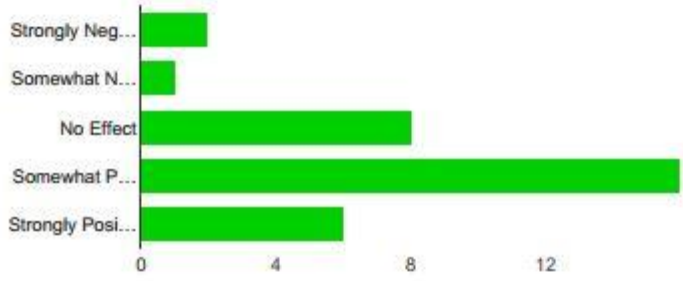
Strongly Negative	2	6.1%
Somewhat Negative	4	12.1%
No Effect	19	57.6%
Somewhat Positive	6	18.2%
Strongly Positive	2	6.1%

↑ Num of Tables in Database [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



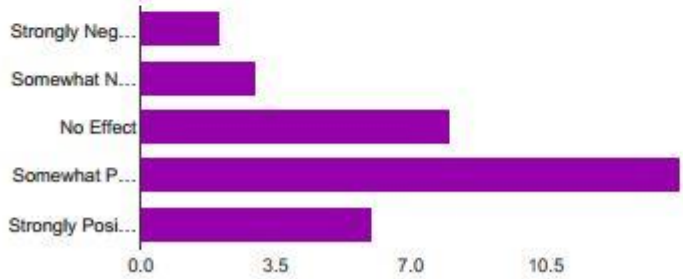
Strongly Negative	3	9.1%
Somewhat Negative	4	12.1%
No Effect	7	21.2%
Somewhat Positive	14	42.4%
Strongly Positive	5	15.2%

↑ Depth of Inheritance Tree [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



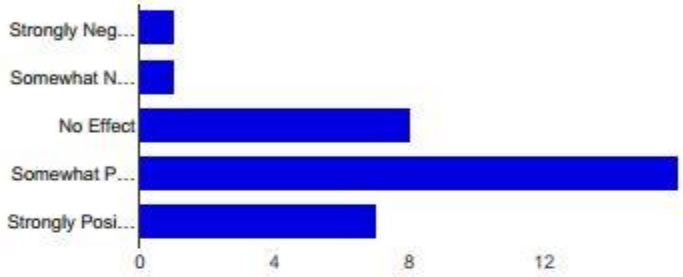
Strongly Negative	2	6.1%
Somewhat Negative	1	3%
No Effect	8	24.2%
Somewhat Positive	16	48.5%
Strongly Positive	6	18.2%

↑ Degree of Coupling [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



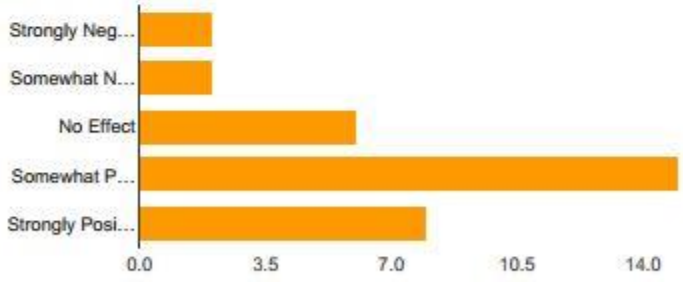
Strongly Negative	2	6.1%
Somewhat Negative	3	9.1%
No Effect	8	24.2%
Somewhat Positive	14	42.4%
Strongly Positive	6	18.2%

↑ Degree of Cohesion [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



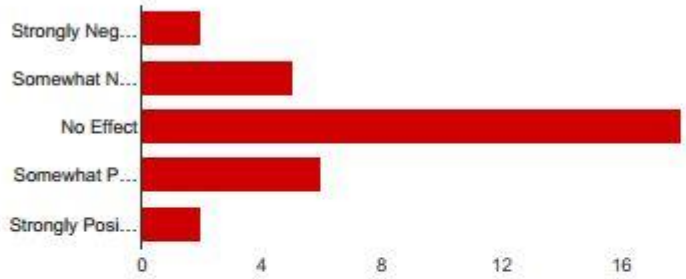
Strongly Negative	1	3%
Somewhat Negative	1	3%
No Effect	8	24.2%
Somewhat Positive	16	48.5%
Strongly Positive	7	21.2%

↑ Num of Attributes [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



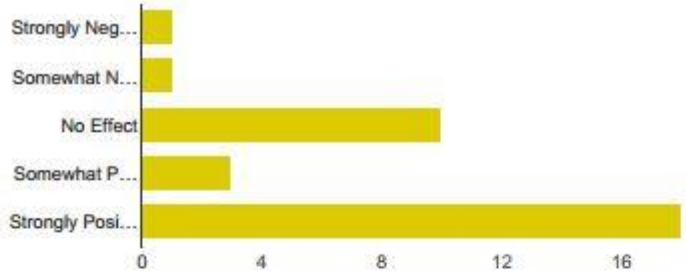
Strongly Negative	2	6.1%
Somewhat Negative	2	6.1%
No Effect	6	18.2%
Somewhat Positive	15	45.5%
Strongly Positive	8	24.2%

↑ Number of Methods [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



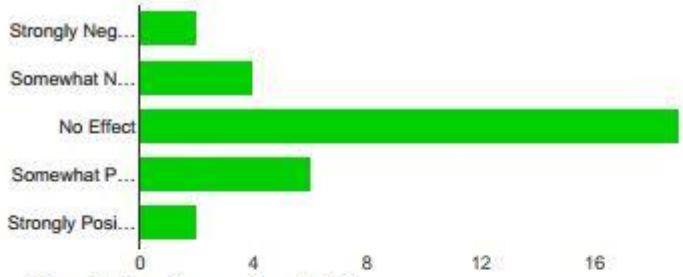
Strongly Negative	2	6.1%
Somewhat Negative	5	15.2%
No Effect	18	54.5%
Somewhat Positive	6	18.2%
Strongly Positive	2	6.1%

↑ Number of Intents [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



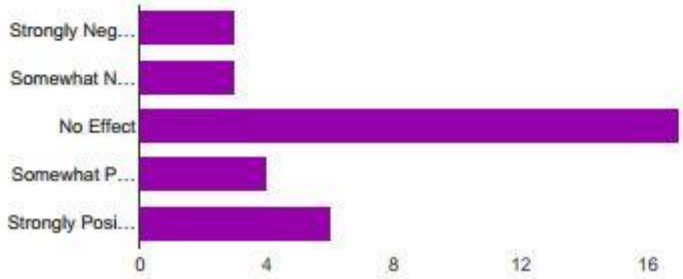
Strongly Negative	1	3%
Somewhat Negative	1	3%
No Effect	10	30.3%
Somewhat Positive	3	9.1%
Strongly Positive	18	54.5%

↑ Cyclomatic Complexity [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



Strongly Negative	2	6.1%
Somewhat Negative	4	12.1%
No Effect	19	57.6%
Somewhat Positive	6	18.2%
Strongly Positive	2	6.1%

↑ Num of <uses-permission> [12- Select the degree of effects for each factors on the COMPATIBILITY of Mobile Application.]



Strongly Negative	3	9.1%
Somewhat Negative	3	9.1%
No Effect	17	51.5%
Somewhat Positive	4	12.1%
Strongly Positive	6	18.2%

What other factors would you offer that affects COMPATIBILITY?

Appendix D: Cronbach's Alpha Values of Survey Results

```
RELIABILITY
  /VARIABLES=VAR00001 VAR00002 VAR00003 VAR00004 VAR00005 VAR00006 VAR00007
 VAR00008 VAR00009
    VAR00010 VAR00011 VAR00012 VAR00013 VAR00014 VAR00015 VAR00016 VAR00017
 VAR00018
  /SCALE('ALL VARIABLES') ALL
  /MODEL=ALPHA
  /STATISTICS=DESCRIPTIVE SCALE CORR
  /SUMMARY=TOTAL.
```

1. FUNCTIONAL SUITABILITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,758	,768	18

2. RELIABILITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,835	,834	18

3. PERFORMANCE EFFICIENCY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,707	,773	18

4. PORTABILITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,818	,823	18

5. MAINTAINABILITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,867	,869	18

6. DATA INTEGRITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,880	,883	18

7. USABILITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,871	,875	18

8. SECURITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,805	,819	18

9. COMPATIBILITY

Reliability Statistics

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
,897	,903	18

TEZ FOTOKOPİ İZİN FORMU

ENSTİTÜ

Fen Bilimleri Enstitüsü ☐

Sosyal Bilimler Enstitüsü ☐

Uygulamalı Matematik Enstitüsü ☐

Enformatik Enstitüsü ☐

Deniz Bilimleri Enstitüsü ☐

YAZARIN

Soyadı : .....
Adı : .....
Bölümü : .....

TEZİN ADI (İngilizce) : .....
.....
.....
.....
.....

TEZİN TÜRÜ : Yüksek Lisans ☐ Doktora ☐

1. Tezimin tamamı dünya çapında erişime açılsın ve kaynak gösterilmek şartıyla tezimin bir kısmı veya tamamının fotokopisi alınsın. ☐
2. Tezimin tamamı yalnızca Orta Doğu Teknik Üniversitesi kullancılarının erişimine açılsın. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.) ☐
3. Tezim bir (1) yıl süreyle erişime kapalı olsun. (Bu seçenekle tezinizin fotokopisi ya da elektronik kopyası Kütüphane aracılığı ile ODTÜ dışına dağıtılmayacaktır.) ☐

Yazarın imzası ..... Tarih .....

RB-SA01/F01 Rev:0 26.10.2011