PRIORITIZING USE CASES FOR DEVELOPMENT OF MOBILE APPS USING AHP: A CASE STUDY IN TO-DO LIST APPS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
ATILIM UNIVERSITY

BY

ONUR YILDIRIM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
THE DEPARTMENT OF SOFTWARE ENGINEERING

SEPTEMBER 2019

Approval of the Graduate School of Natural and Applied Sciences, Atilim University.

____

Prof. Dr. Ali KARA
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of **Master of Science in Software Engineering, Atilim University**.

____

Prof. Dr. Ali YAZICI
Head of Department

This is to certify that we have read the thesis Prioritizing Use Cases For Development Of Mobile Apps Using Ahp: A Case Study In To-Do List Apps submitted by Onur YILDIRIM and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

__________________________	__________________________
Prof. Dr. Ali YAZICI Co-Supervisor	Asst. Prof. Dr. Serhat PEKER Supervisor

Examining Committee Members:

Asst. Prof. Dr. Çiğdem TURHAN
Software Eng. Department, Atilim University __________

Asst. Prof. Dr. Serhat PEKER
Dept. of Management Information Systems, İzmir Bakırçay University __________

Asst. Prof. Dr. Bilge SAY
Software Eng. Department, Atilim University __________

Asst. Prof. Dr. Güzin TİRKEŞ
Computer Eng. Department, Atilim University __________

Asst. Prof. Dr. Ebru GÖKALP
Dept. of Technology and Knowledge Management, Başkent University __________

Date: *10.09.2019*

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.

Onur YILDIRIM

ABSTRACT

PRIORITIZING USE CASES FOR DEVELOPMENT OF MOBILE APPS USING AHP: A CASE STUDY IN TO-DO LIST APPS

Yıldırım, Onur
M.S., Department of Software Engineering
Supervisor : Asst. Prof. Dr. Serhat PEKER
Co-Supervisor : Prof. Dr. Ali YAZICI

September 2019, 33 pages

The intensity of use of mobile applications has increased significantly with the rapid development of communication technologies. Every day many different types of To-Do apps are uploaded to mobile application markets. However, it is very difficult for the apps to stay competitive and survive in these marketplaces. One of the success factors in the mobile application market is the functionality of mobile applications. Identifying the functions of the application is very important for mobile application developers. In this manner, this study aims to prioritize to the functionalities of mobile apps. For this purpose, Analytic Hierarchy Process (AHP) is used to assess the use cases for the development of mobile apps. The results of case study applied in to-do list apps show that AHP can be used as an efficient tool to determine the importance of the requirements in mobile apps.

Keywords: Analytic Hierarchy Process, Use case Prioritization, Requirements Prioritization, Mobile Application, To-Do List Apps.

ÖZ

AHP KULLANILARAK MOBİL UYGULAMALARIN GELİŞTİRİLMESİ İÇİN KULLANIM DURUMLARINI ÖNCELİKLENDİRME: YAPILACAKLAR LİSTESİ UYGULAMALARINDA BİR VAKA ÇALIŞMASI

Yıldırım, Onur
Yüksek Lisans, Yazılım Mühendisliği Bölümü
Tez Yöneticisi : Dr. Öğr. Üyesi Serhat PEKER
Ortak Tez Yöneticisi : Prof. Dr. Ali YAZICI

Eylül 2019, 33 sayfa

Mobil uygulamaların kullanım yoğunluğu, iletişim teknolojilerinin hızlı gelişimi ile bağlantılı olarak önemli ölçüde artmıştır. Her gün pek çok farklı Yapılacaklar listesi uygulaması mobil uygulama pazarlarına yüklenmektedir. Ancak, uygulamaların rekabetçi kalması ve bu pazarlarda hayatta kalması çok zordur. Mobil uygulama pazarındaki başarı faktörlerinden biri mobil uygulamaların işlevselliğidir. Uygulamanın işlevlerini doğru tanımlamak, mobil uygulama geliştiricileri tarafından pazar gücünü etkiler. Böylece, bu çalışma mobil uygulamaların işlevlerine öncelik vermeyi amaçlamaktadır. Bu amaçla, mobil uygulamaların geliştirilmesine yönelik kullanım durumlarını değerlendirmek için Analitik Hiyerarşi Süreci (AHP) kullanılır. Yapılacaklar listesi uygulamalarında uygulanan durum incelemesinin sonuçları, AHP'nin mobil uygulamalardaki gereksinimlerin önemini belirlemek için etkili bir araç olarak kullanılabileceğini göstermektedir.

Anahtar Kelimeler: Analitik Hiyerarşi Süreci, Kullanım Durumları Önceliklendirme, Gereksinim Önceliklendirme, Mobil Uygulama, Yapılacaklar Listesi Uygulamaları.

To my mother, my wife and my daughter

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my supervisor Dr. Serhat Peker for his invaluable guidance, patient, trust, encouragements, advices and insight during the entire study.

I am also grateful to Prof. Dr. Ali Yazıcı for his valuable comments and guidance.

I shall also thank to my thesis jury members Dr. Bilge SAY, Dr. Çiğdem TURHAN, Dr. Güzin TİRKEŞ and Dr. Ebru GÖKALP for their advice and guidance in helping to shape this dissertation.

Sincere thanks to my family, especially my dear mother Nurhan Keçeci for her love and support.

Finally, I would like to present my special thanks to my dearest wife, Tuğba YILDIRIM, for her love, support, patience, encouragements and being with me whenever I need since the beginning of my Master Program.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS/ABBREVIATIONS

AHP	Analytic Hierarchy Process
FAHP	Fuzzy Analytic Hierarchy Process
CR	Consistency Ratio
CI	Consistency Index
RI	Random Index
MSNI	Mon Space Net Incorporated
SSD	System String Scheme
IES	The Institute Exam System
CRUD	Create, Read, Update, Delete
UC	Use case

CHAPTER 1

INTRODUCTION

In recent years, smart phones and tablets have been used as an assistant for people's daily life. Mobile applications are common and popular for mobile devices, and people need to use with these applications. There are many applications in the market. People can easily upload apps to markets like Google Play or Apple App Store and many applications continue to be uploaded and downloaded daily. Therefore, users will rely more on mobile applications they use for features such as messaging or purchasing for applications [1] [2]. Digital markets have evolved into popular platforms for mobile applications and application developers can develop applications that can reach millions in numbers [3] [4]. With the rapid rise and development of mobile service systems in the world in recent years, the desire to find new customers provides a significant source of income for mobile service providers, mobile network operators, mobile content and application developers. With the rapid growth in the number of users, a significant increase in the number of mobile content / implementation developers is also seen. The mobile application market has become a rapidly growing power thanks to the intense interest of users to the mobile applications. When the mobile applications and computer applications are compared, the mobile applications are rapidly developing, growing and the cost reducing, the ease of application development and access to rising sector. Understanding the importance of requirements in practice is often a difficult process. Many applications in terms of review quality and user comments may show weakness [6].

Even though, many market applications faced difficulties. Developers and companies are always in competition with each other. It is very difficult to reach considerable number of users, so many developers and companies going to fail. Customer needs and satisfaction must be the most important focus when evaluating. In a complex situation, customer needs are important in making the right decision. The misunderstanding of the requirements may reach out serious consequences on the development process. In order to take the lead in this race, the companies should consider these situations and choose the right use cases for the development of applications to the least damage [7].

The systematic evaluation of the needs and objectives of the stakeholders should be carried out with appropriate feature selection and feature model customization. Moreover, user's satisfaction is very important so quality and cost affect choice of applications from users.

Therefore, in order to produce high quality mobile apps and to ensure customer satisfaction, it is necessary to know the needs of customers, define their wishes and try to do the best to satisfy them. For this purpose, it is very important to determine and prioritize application requirements by users. In this manner, Analytic Hierarchy Process (AHP) is a multi-criteria decision-making method by comparing pairwise factors.

1.1. Purpose of the Study

This study aims to evaluate and prioritize use cases for mobile apps using Analytic Hierarchy Process (AHP) technique. For this purpose, a case study is conducted in to-do list apps and results are interpreted.

1.2. Significance of the Study

In this study we implement the use of Analytic Hierarchy Process (AHP) to rank the importance of use cases in the application of mobile applications. This study provides insights for researchers by showing that Analytic Hierarchy Process (AHP) is applicable to prioritize use cases in domain of mobile apps.

1.3. Thesis Organization

The remainder of this study is structured as follows. In Section 2, literature survey is provided. Section 3 describes methodology and analyses. Also, results carried out for to-do list app category is presented. Finally, Section 4 presents concluding remark.

CHAPTER 2

LITRERATURE SURVEY

2.1. Requirement Prioritization

Requirement prioritization has many techniques in software engineering field. When developing software, it is easier to prioritize the requirements, rather than examine everything as a whole. [9]. One of the oldest software in systematic literature scans may be requirement prioritization. Previous studies have shown that eight selected computational requirement prioritization techniques were discussed in an objective way for compare. In most of the mentioned requirement prioritization techniques, it has been found that only the small-scale needs cluster is directly related. At the first stage, it should be decided which requirements are applied and the elements to be postponed or continued in the next process. Researchers, scientists and academicians have already used these techniques and they have been able to achieve some results. In a comprehensive review of the requirement prioritization [10], existing requirement prioritization techniques and limitations of stakeholders involved in software development contexts in the system development process determine the future of requirement prioritization. Based on the developmental stage of the decision-making process, the rationale for evaluation alone is insufficient. In addition, how the connection between them is determined in practice affects the decision-making process. There are many factors associated with complex problems. In the measurements, the elements are determined by measuring each one by one. In [11], Simpson's 1/3 rule, Simpson's 3/8 rule, trapezoid rule ... etc. have applied the rules. With the mini software (digital integration, MSNI) developed to find the software requirements, the existing techniques are more easily applied based on the value of the given functions. In [12] developed a systematic mapping method specific to requirement prioritization software for experimental studies, and when we look at the basis of the studies, they mentioned that they are doing research to emphasize requirement prioritization and technical methods of experimental research. The classification was used to determine the type of the research indicated in [13]. Validation research, classification, evaluation researches, solution proposal documents, opinion documents, philosophical documents and experiential documents are seen to be separated. The only difference in this situation is to highlight the examination documents instead of philosophical documents. A comprehensive review of the requirement prioritization is the category of review papers. The reason for choosing this category is the lack of philosophical documents. Dividing papers into multiple categories can enable the mapping to become operational.

In [14], researchers applied web surveys for eliciting the software requirements. In the dissemination of requirement prioritization techniques, current analysis of current studies is applied. For researchers and practitioners, the method of analysis improves presence. The best analysis of this time has been done and 49 techniques have been analyzed. The properties and analyses of the conditions of the requirement prioritization were analyzed. In the software development method, the importance of requirement prioritization was evaluated, and a new perspective was determined. Experimental methods for previously unapplied requirement prioritization techniques and constraints were developed in [10]. The measurement scales were not based on investigations and restrictions. He argued that there were difficulties with time consumption, requirements and scaling.

2.2. Use Case Prioritization

Use case and activity diagrams were about a new approach to prioritization. The researchers developed new approach which functions include creating and prioritizing a test scenario. In [15] divided the prioritization of use cases into four parts. The first step has identified actor priority as prioritizing the communication between the actor and the user. The second step is to calculate the use cases priority from the use cases diagrams and the highest value use case has the highest priority. The third step is Customer input to achieve customer priority in case of use and to understand which use case is more important for the customer. And the last step is to calculate the use case priority. It is prioritized considering the technical requirements. There are a few studies in determining priority of use cases and there are not enough literature review in this context. In [16] recommends a skeleton for prioritization scenarios and their use cases. Within the infrastructure of the software development cycle, there are requirements stages. In this context, business objectives, relations between use cases and business objectives are subjected to a sample and priorities of use cases are determined. There are two levels of recommended areas. First, it is used to determine the importance of use in the model use case scenario and, in some cases, to determine if it is successful and what effect the system functions have on targets. Second, it determines the priority in use cases based on measurements such as the total number of objects in each step of the use cases. [17], in the modelling of use cases, a set of factors were used to determine the order of priorities in structural modelling and specific frames were proposed. Systematic functionality based on criticality in sorting is easy to determine and features such as error tolerance and quality assurance can be considered. It is first used for critical system levels for use cases and then for lower case use cases. This process is repeated until a sequential use case is created [18].

In [16] proposed that another use case prioritization method. They suggested approaches to use case scenarios based on density. In the first instance, a System String Scheme (SSD) is created in connection with the scenario, followed by a graphical representation based on the system layout schema, and in the final step, priority is given to the situation linked to the scenario graph. Each step in the Scenario Graph represents the use case. In the Scenario Graph, the first depth search is applied to the tree use case structure. The next step is to assign weight to each use case in relation to the formulation. It is an easy and efficient approach for Stratified AHP that increases and organizes expertise and product range of the model. Its main benefits are its simplicity, practicality, and its ability to accommodate the desired use cases. In addition, thanks to its simplicity, it is easy and inexpensive to develop the approach. Stratified AHP proposes a two-dimensional comparison rather than the use case where the scale is represented by a single value when comparing two samples [19]. In [20], As a suggestion for software priority approximations, centrism is shown based on the complexity in case of probability. Approaches can be applied in the first stage for software requirements. In use case prioritization, analysts and software developers need to further develop the facts or consider the features that are important to be able to benefit from the development of a specific software. In order to determine the order of priority, the method of measurement for the network node center is applied to calculate the method of production from feature prioritization and the prioritization to the software requirements. Validation is determined by applying the three recommended real model models. In [21], The approach is completely independent of the analytical and the use case approach and independent of the system analyst's contribution. In the use case template, it is possible to determine the data by determining the cost measure for each use case. On the other hand, an enough test input can be applied to test according to the cost scenario. Moreover, even a single error in the scenario cost will have a large impact on system reliability in high-scenario cost measurement. If the design steps of the approach have not been initiated, an early scenario cost value can be calculated for the software development life cycle. Therefore, based on scenario cost measurement, use cases such as reducing matches and increasing compliance status are shown more in order to improve software quality. They defines the critical components of the system and proposes a better software method.

2.3. Analytic Hierarchy Process

Analytic Hierarchy Process is a complex decision-making method involving personal judgment. The Analytic Hierarchy Process method, which is more accurate and powerful, has been chosen where the old methods are weak in complex decision making. It is known that personal judgment and opinions do not find a clear solution in complex problems. Until recently, it was very difficult to find a solution before the Analytic Hierarchy Process method in decisive definitions and linguistic differences [22]. In Analytic Hierarchy Process, problems proceed for a purpose and form a model that lists the levels of elements that influence the goal. At this stage, the problems are divided into hierarchy and sub-sections. Thus, it examines each element in more detail. Analytic Hierarchy Process aims to make judgments about decision-making problems and to make the best of alternative findings by evaluating them according to preference, importance and probability order [23]. Analytic Hierarchy Process is used to determine the information of decision makers, it cannot fully express the traditional thinking structure [24]. Analytic Hierarchy Process is used to solve multifaceted and layered problems in decision making methods. The level of knowledge and uncertainty in people's reasoning resembles fuzzy set theory in complex decision-making methods. The most important contribution of fuzzy set theory is to reveal uncertainty. The combination of Analytic Hierarchy Process and fuzzy set theory ensures that the multi-faceted decision-making process is effectively demonstrated [25]. The oldest known study is the comparison of the fuzzy proportions of [26]. It is the method we always need for research in decision-making processes. Many Analytic Hierarchy Process studies have been proposed to calculate weights in the assessment [27] [28] [29]. The preferred reason for the fuzzy Analytic Hierarchy Process approach is to reveal the evaluation of decisions that have not been determined. When comparing the traditional Analytic Hierarchy Process methods, a certain ratio scale is used. The scale, which carries simplicity and ease of use, cannot define ambiguity with its opinion or evaluation relationship. Moreover, it is a known fact that evaluations about the relevance of personal needs do not always yield subjective and precise results. Fuzzy Analytic Hierarchy Process (FAHP) method was found to be less time consuming and easier to use than other methods [30].

In [16] prioritize the use cases, they have been conducted a new empirical study of the use of neuro-fuzzy systems. In this study, Analytic Hierarchy Process is used with fuzzy models to use case priorities. Prioritization studies quality, usability, satisfaction, efficiency, availability, price, performance, effectiveness, acceptance, security ... etc. are the general criteria for prioritization studies. However, prioritization studies may vary according to the type of research conducted. For example, battery life for smartphones, display quality or resolution for monitors, academic skills of academics, and these can be further increased. The issues that are predominantly selected by the users are weighted up and the less selected subjects are not taken into consideration [31] [32].

In [9], Analytic Hierarchy Process provided scalability, computational complexity, ease of use, reliability of results, and limitation of strategies, requirements, and dependency issues in the business environment by improving existing prioritization techniques. Analytic Hierarchy Process is a measurement theory that predicts the path of binary comparisons. The decisions of the experts are used to obtain priority scales and they measure the intangible assets relatively. Comparisons are a judicial scale, which indicates that one element is more than the one given by [33]. In some cases, decisions can be inconsistent, Analytic Hierarchy Process is activated to improve decisions and achieve higher levels of consistency. Some steps are needed to make an Analytic Hierarchy Process in an organized way. In order to compare, there is a need for a number scale which shows how dominant and important an element is compared to the other elements and For the comparison processes, Saaty's nine-point scale is employed [34]. The Analytic Hierarchy Process (AHP) has been tested in a variety of settings to make decisions. When IBM designed the AS 400 computer in 1991, it chose Analytic Hierarchy Process as a process. IBM has won the prestigious Malcolm Baldrige Award for Excellence through this work. Bauer published a statement on how Analytic Hierarchy Process should be compared [23]. Also, AHP has different use cases and techniques. In [35] applied an even-step method to select requirements from requirements groups. These applications are shaped according to the scope of the project to increase the reliability and usability of the result. In [36], "The Institute Exam System (IES)" has adopted a goal-oriented approach in calculating the costs and necessity of needs and the effort spent in the process. By using the Analytic Hierarchy Process, they determined the cost and requirements according to their priority. In [37], FAHP is the first study to analyze user interfaces in the digital library and in an evaluation model. Since the user page criteria are personal and focused on the person and it is judged and integrated with the AHP to make the project where the problems about the decision-making situation are planned. Researchers work in the process which need to target user interfaces and improve quality. In digital library interfaces, FAHP is based on the most important criteria and has high flexibility as well as practical values. In the judgments composed of different types of human types, a statistical procedure should be applied by calculating the alternative weights by calculating the variation between the few individuals [38].

AHP forms the basis of decision-making and is a decision-based decision-making process by dividing the problem into different sections [39]. The primary distinguishing features of the AHP are to convert comparisons based on experimental data to mathematical models with numerical data [40]. The use of AHP in mobile applications has not been widely used in research and articles. In the AHP approach, it would be correct to choose the determinative approach if it consists of groups that interact with each other in the selection of the subject. The lack of this situation has inspired the work of this article. Increasing the use of AHP in this area may lead to more accurate results. In this article, prioritization studies, use and implementation of AHP have been studied.

CHAPTER 3

METHODOLOGY AND ANALYSES

In this study, the main objective is prioritizing the use cases for to-do List mobile applications. Once mobile applications have been identified and 68 mobile applications have been selected, the common use cases in all applications have been selected. The use cases are specific for to-do list and shopping list applications so there should be features such as create, read, list and delete (CRUD) functions and the evaluators need to be selected as priority.

The evaluators of this study are 54 undergraduate software engineering students who were the participants of the requirements engineering course. Also, activation score was given to increase students' motivation. The questionnaires were given to this sample. This sample was selected since they were all knowledgeable about requirements/use case prioritization and AHP usage in the process. This sample is also important for the users of mobile apps. In order to ensure that all evaluators have familiarity with the to-do lists apps, the evaluators were requested to actively use Any.do, which is one of the popular mobile apps in this category, over a week prior to data collection. The possible basic and common features of the to-do list applications were determined and given to the evaluators for evaluation in the questionnaire in Table 1.

Table 1 Use Cases

Use Case ID	Requirement/Functionality
UC1	Create list
UC2	Read/View List with its items
UC3	Update/Rename list
UC4	Delete list (with its items)
UC5	Clear List
UC6	Add an item to list via keyboard
UC7	Add an item to list via voice
UC8	Update a list item
UC9	Delete a list item
UC10	Mark a list item as important
UC11	Set a reminder for a list item
UC12	Set all list items as checked/unchecked
UC13	Set a list item as checked/unchecked
UC14	Move an item from one list to another list
UC15	Share a list
UC16	Share a list item
UC17	Add attachment to a list item
UC18	Add note to a list item

When evaluators finish this process, the data were entered in a MS Excel file which is template of AHP in Table 1. The consistency check for each questionnaire was found and then analyzed the results separately. Hence, Consistency Ratio (CR) was used and then, if the value of evaluators are less than or equal to 0.1, he considered to have consistent responses [23]. The weights of the use cases were calculated using AHP method based on the assessments of respondents who passed the consistency test. By taking the mean of individual evaluator scores, the aggregate values of relative importance weights were calculated. For use cases, the weights calculated sum up to 1, then they were multiplied by 100 so that their aggregate would be 100. Finally, since these scores out of 100, the use case prioritization and all interpretations were performed.

UCs	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16	UC17	UC18
UC1																		
UC2																		
UC3																		
UC4																		
UC5																		
UC6																		
UC7																		
UC8																		
UC9																		
UC10																		
UC11																		
UC12																		
UC13																		
UC14																		
UC15																		
UC16																		
UC17																		
UC18																		

Figure 1 Comparison Matrix

The questionnaire given to the evaluators was given as scales as in the Figure 1 and they were asked to evaluate them according to these scales. It requires a scale for the expression of the reciprocal rating in the pairwise comparison and is consistent with the main purpose of the prioritization order. We can easily find the significance of all features between them each other. Therefore, the scale is used to express individual preferences for comparison [34].

Table 2 Importance levels for AHP

Fundamental Scale	**Importance**
Extremely less important	1/9
	1/8
Very strongly less important	1/7
	1/6
Strongly less important	1/5
	1/4
Moderately less important	1/3
	1/2
Equal important	**1**
	2
Moderately more important	3
	4
Strongly more important	5
	6
Very strongly more important	7
	8
Extremely more important	9

After the determination of mobile applications features, AHP study of the features in to-do list applications were completed with the students. In the given document for students, there are importance level whereas comparing features. The experiment form has been prepared in Table 2. In addition to this, experiment form has requirements list, description of study and the blank Pairwise Comparison Matrix is provided for filling.

The values used in the comparison of each feature are numerical values, which are "extremely more important" is equal to 9, "equally important" is equal to 1 and "extremely less important" is equal to 1/9 [41]. This scale converts to a numerical measure showing the relative importance of comparative substances corresponding to each judgement.

3.1 AHP Methodology

We employed four-stage approach to prioritize use cases for to-do List mobile applications, see Figure 2. Firstly, definition of the use cases for the applications which is selected by the most regular. Second stage is comparing the use cases with survey. Third stage is to use Analytic Hierarchy Process (AHP) method for calculating the importance weights of the use cases and rank them. The last stage is to sum up the total weights of the use cases.

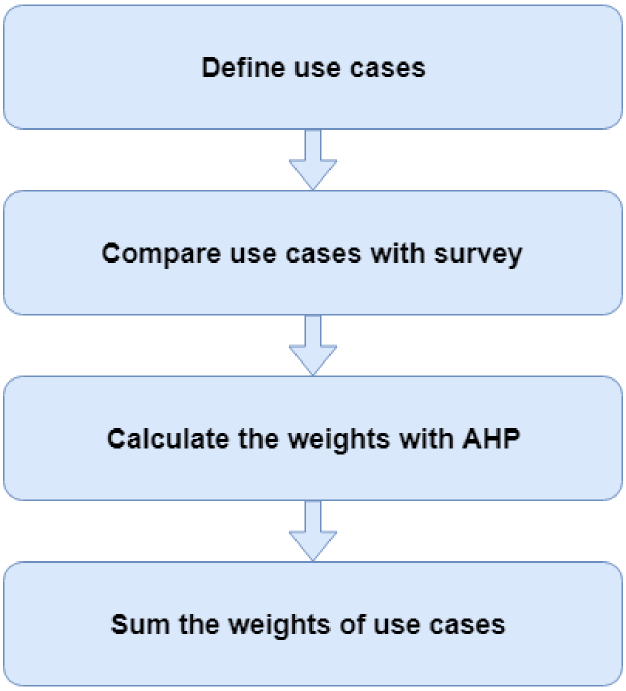

Figure 2 Stages of Study

Stage 1- Define the use cases for the selected mobile application domain. The most common functions of to-do lists and shopping list applications are selected. These features are common in most applications.

Stage 2- Compare the use cases according to the survey. It was necessary to form pairwise comparison matrices in order to determine the importance of all elements relative to each other. The matrices were determined by comparing them in pairwise between the criteria or options of the decision maker. It was decided whether the structure of the use cases was appropriate for AHP. The definitions were determined in a specific order of priority. The highest limit in the hierarchy is the main target and the lowest limit was the alternative. In order to better identify the use cases, they must separate them into components and organize each component in a hierarchical system. If these elements have features that can affect the main target, different steps can be added to the hierarchy. For each use case, a structured hierarchy model was established from the objective, criteria and sub-criteria levels. When creating the hierarchy, it was assumed that the same level of components was independent of each other.

Stage 3- Calculate the weights with AHP. Once the pairwise comparison matrices are created and their consistency is determined, the values of the compared use cases are found. For the calculation of the priority vectors of all use cases, the sum of each row in the pairwise comparison matrix is calculated and divided by the sum of all rows. Calculations are made by taking the sum of the evaluations made in the normalization calculation.

Stage 4- Sum up the weights of use cases. The priority values of the use cases are summed up and divided together, the use cases are sorted from the highest to the lowest.

The Analytic Hierarchy Process (AHP) is a multi-layered decision-making method to solve complex decision-making problems. Sort multiple options are in order of importance by specific criteria. AHP was established in a systematic structure where decision makers could be involved. AHP, it was revealed in 1974 by Saaty [34]. Criteria that can be evaluated are quantitative and qualitative, but also a decision-making method that includes humanitarian judgment and criticism.

Based on alternative benchmarks, the AHP offers problems in a hierarchical structure at multiple levels. A classifier structure is used to determine the purpose, criteria, and options of the problem [23]. AHP helps the decision maker to solve multi-criteria and layered decision problems. As an evaluation criterion, AHP is determined as an alternative option to choose the right decision. In some cases, there are contradictions in the criteria, so it is not the most appropriate of the right choice. AHP forms a weight in comparison to the criteria of the decision-maker in the options it evaluates. For fixed options, points are awarded based on the comparison of the options based on the criteria of the decision makers. The higher the number of points, the better the performance of the option. The sum of the criteria weights and the points of the score are determined by an overall score system for each option.

AHP is one of the most common decision-making methods used for complex systems [42]. The comparison results applied in this method are defined from 1 to 9 for each pair of factors. The higher number of comparisons are compared with other factors. For the calculation of the weights of the options, the sum of each number column in matrix A is taken and divided. The relative weight of each option is then the average for each line.

$$A = \begin{bmatrix} a_1 & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{bmatrix}$$

The relative significance weight is calculated by taking the average of the Eigen vector. Eigen vectors are those vectors when a linear transformation (AHP comparison matrix) is performed on them then their direction do not change. The sum of all points of the priority vector is equal to 1. The consistency index (CI) is used to calculate the degree of consistency [43].

$$CI = (\lambda_{max} - n) \; / \; (n - 1)$$

The number of criteria is expressed in n. In the ƛ_max is equal to the eigenvalue of matrix A. After making binary comparison matrices, weight vectors are calculated first. One of the important point is to determine the suitability of the desired direction. Formulation calculations are carried out to determine the eigenvectors of the matrices with the largest eigenvalues. Therefore, priority and order of eigenvalues and eigenvectors are determined. The greatest eigenvalue of matrix A is consistent with ƛ_max being equal to n [23]. The highest eigenvalue ƛ_max is calculated as shown below [44].

$$\lambda_{max} = \sum_{i=1}^{n} \frac{(c.v)_i}{n.v_i}$$

In the formula, "c" represents the pairwise comparison matrix, and "v" represents the matrix vector. " CI " is the consistency index value. When this value is obtained, it can be calculated by dividing by " RI " the random index in Table 3 [45].

Table 3 Values of RI

n	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
IR	0	0	0,58	0,9	1,12	1,24	1,32	1,41	1,45
n	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**
IR	1,49	1,51	1,54	1,56	1,57	1,58	1,59	1,6	1,61

$$CR = \frac{CI}{RI}$$

Values of up to 10% are taken for acceptance of the consistency rate. If these values exceed 10 (C.R> 0.10), inconsistency is detected. It must be checked again until it is 10% or less [44].

3.2. AHP Implementation

One of the most practical ways of using the AHP method is to divide complex problems into simple components. A large number of the use cases make the decision-making process very complex. When problems are divided in the process, the solution is approached more easily. In this case, the problem is divided by the pairwise comparison method. Furthermore, it is not possible to ensure the relative weight of the relative weights if no pairwise comparison is made. It is used to obtain the pairwise comparison decision matrix for calculating the weights of each criteria received and ranks them among themselves.

54 software engineering undergraduate students participated in the study and 41 of them had CR scores less than 15% and 34 of them were less than 10%. Criteria were compared until the consistency degree of the decision matrix was confirmed.

	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16	UC17	UC18
UC1	1	3	1	1	3	3	9	1	1	2	1	5	3	9	9	9	5	2
UC2	1/3	1	1/3	1/3	1	1	3	1/3	1/3	1/2	1/3	2	1	3	3	3	2	1/2
UC3	1	3	1	1	3	3	9	1	1	2	1	5	3	9	9	9	5	2
UC4	1	3	1	1	3	3	9	1	1	2	1	5	3	9	9	9	5	2
UC5	1/3	1	1/3	1/3	1	1	3	1/3	1/3	1/2	1/3	2	1	3	3	3	2	1/2
UC6	1/3	1	1/3	1/3	1	1	3	1/3	1/3	1/2	1/3	2	1	3	3	3	2	1/2
UC7	1/9	1/3	1/9	1/9	1/3	1/3	1	1/9	1/9	1/5	1/9	1/2	1/3	1	1	1	1/2	1/5
UC8	1	3	1	1	3	3	9	1	1	2	1	5	3	9	9	9	5	2
UC9	1	3	1	1	3	3	9	1	1	2	1	5	3	9	9	9	5	2
UC10	1/2	2	1/2	1/2	2	2	5	1/2	1/2	1	1/2	3	2	5	5	5	3	1
UC11	1	3	1	1	3	3	9	1	1	2	1	5	3	9	9	9	5	2
UC12	1/5	1/2	1/5	1/5	1/2	1/2	2	1/5	1/5	1/3	1/5	1	1/2	2	2	2	1	1/3
UC13	1/3	1	1/3	1/3	1	1	3	1/3	1/3	1/2	1/3	2	1	3	3	3	2	1/2
UC14	1/9	1/3	1/9	1/9	1/3	1/3	1	1/9	1/9	1/5	1/9	1/2	1/3	1	1	1	1/2	1/5
UC15	1/9	1/3	1/9	1/9	1/3	1/3	1	1/9	1/9	1/5	1/9	1/2	1/3	1	1	1	1/2	1/5
UC16	1/9	1/3	1/9	1/9	1/3	1/3	1	1/9	1/9	1/5	1/9	1/2	1/3	1	1	1	1/2	1/5
UC17	1/5	1/2	1/5	1/5	1/2	1/2	2	1/5	1/5	1/3	1/5	1	1/2	2	2	2	1	1/3
UC18	1/2	2	1/2	1/2	2	2	5	1/2	1/2	1	1/2	3	2	5	5	5	3	1

Figure 3 Decision Matrix of an Evaluator

Each column obtained was collected and then the values were divided by the column sum to obtain the decision matrix like Figure 3. The relative weight of each section was equal to the sum of the sequence values. The matrix eigenvector was multiplied by the relative weights of each layer and the decision matrix and the results were collected and calculated.

Table 4 Weights of an Evaluator and the Results of Consistency

λ_{max}	18,04674
CI	0,002749
RI	1,61
CR	0,0017077

# of UC	Weights
1	10,80%
2	3,60%
3	10,80%
4	10,80%
5	3,60%
6	3,60%
7	1,20%
8	10,80%
9	10,80%
10	6,00%
11	10,80%
12	2,10%
13	3,60%
14	1,20%
15	1,20%
16	1,20%
17	2,10%
18	6,00%

According to this information, the inputs of the evaluators are listed. The AHP method was applied to determine the paired comparison matrix and relative weights. The CI decision of the first evaluator was 0.002749, the CR value was 0.0017077 and the RI value was 1.61 as show Table 4. This matrix was found to be consistent in the light of results.

	U C1	U C2	U C3	U C4	U C5	U C6	U C7	U C8	U C9	UC 10	UC 11	UC 12	UC 13	UC 14	UC 15	UC 16	UC 17	UC 18
UC 1	1	2	2	3	3	4	4	5	5	6	7	9	8	7	8	8	9	9
UC 2	1/2	1	1	2	2	3	3	4	4	5	6	8	7	6	7	7	8	8
UC 3	1/2	1	1	2	2	3	3	4	4	5	6	8	7	6	7	7	8	8
UC 4	1/3	1/2	1/2	1	1	2	2	3	3	4	5	7	6	5	6	6	7	7
UC 5	1/3	1/2	1/2	1	1	2	2	3	3	4	5	7	6	5	6	6	7	7
UC 6	1/4	1/3	1/3	1/2	1/2	1	1	2	2	3	4	6	5	4	5	5	6	6
UC 7	1/4	1/3	1/3	1/2	1/2	1	1	2	2	3	4	6	5	4	5	5	6	6
UC 8	1/5	1/4	1/4	1/3	1/3	1/2	1/2	1	1	2	3	5	4	3	4	4	5	5
UC 9	1/5	1/4	1/4	1/3	1/3	1/2	1/2	1	1	2	3	5	4	3	4	4	5	5
UC 10	1/6	1/5	1/5	1/4	1/4	1/3	1/3	1/2	1/2	1	2	4	3	2	3	3	4	4
UC 11	1/7	1/6	1/6	1/5	1/5	1/4	1/4	1/3	1/3	1/2	1	1/2	1/3	1	2	2	3	3
UC 12	1/9	1/8	1/8	1/7	1/7	1/6	1/6	1/5	1/5	1/4	2	1	1/2	1/3	2	2	3	3
UC 13	1/8	1/7	1/7	1/6	1/6	1/5	1/5	1/4	1/4	1/3	3	2	1	1/2	2	2	3	3
UC 14	1/7	1/6	1/6	1/5	1/5	1/4	1/4	1/3	1/3	1/2	1	3	2	1	1/2	1/2	2	2
UC 15	1/8	1/7	1/7	1/6	1/6	1/5	1/5	1/4	1/4	1/3	1/2	1/2	1/2	2	1	1	2	2
UC 16	1/8	1/7	1/7	1/6	1/6	1/5	1/5	1/4	1/4	1/3	1/2	1/2	1/2	2	1	1	2	2
UC 17	1/9	1/8	1/8	1/7	1/7	1/6	1/6	1/5	1/5	1/4	1/3	1/3	1/3	1/2	1/2	1/2	1	2
UC 18	1/9	1/8	1/8	1/7	1/7	1/6	1/6	1/5	1/5	1/4	1/3	1/3	1/3	1/2	1/2	1/2	1/2	1

Figure 4 Decision Matrix of an Evaluator

Table 5 Weights of Evaluator and the Results of Consistency

λ_{max}	19,64828
CI	0,096958
RI	1,61
CR	0,0602221

# of UC	Weights
1	17,60%
2	12,80%
3	12,80%
4	9,10%
5	9,10%
6	6,50%
7	6,50%
8	4,60%
9	4,60%
10	3,30%
11	2,00%
12	1,80%
13	2,10%
14	1,90%
15	1,60%
16	1,60%
17	1,10%
18	1,00%

According to the results in Figure 4, the CI decision of the evaluator II was 0.096958, the CR value was 0.0602221 and the RI value was 1.61 as show Table 5. This matrix was found to be consistent in the light of results.

	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC10	UC11	UC12	UC13	UC14	UC15	UC16	UC17	UC18
UC1	1	4	5	5	4	6	7	4	4	5	6	5	4	7	8	9	6	7
UC2	1/4	1	3	3	2	5	4	2	2	3	4	3	1	5	6	7	4	5
UC3	1/5	1/3	1	1	1/2	4	3	1/2	1/2	3	4	2	2	4	6	7	4	5
UC4	1/5	1/3	1	1	3	5	4	1/2	1/3	2	3	3	1/3	4	5	6	3	4
UC5	1/4	1/2	2	1/3	1	3	2	1/3	1/4	1/3	1/2	1/3	1/3	3	4	5	2	3
UC6	1/6	1/5	1/4	1/5	1/3	1	1/2	1/4	1/5	1/2	2	1/2	1/3	3	4	5	3	2
UC7	1/7	1/4	1/3	1/4	1/2	2	1	1/3	1/4	1/2	1/3	1/2	1/4	2	3	4	1/2	1
UC8	1/4	1/2	2	2	3	4	3	1	1	3	4	3	2	5	6	7	4	5
UC9	1/4	1/2	2	3	4	5	4	1	1	2	3	2	1	4	5	6	4	5
UC10	1/5	1/3	1/3	1/2	3	2	2	1/3	1/2	1	3	2	1/2	4	5	6	3	4
UC11	1/6	1/4	1/4	1/3	2	1/2	3	1/4	1/3	1/3	1	1/2	1/3	2	3	4	1	2
UC12	1/5	1/3	1/2	1/3	3	2	2	1/3	1/2	1/2	2	1	1/2	3	4	5	2	3
UC13	1/4	1	1/2	3	3	3	4	1/2	1	2	3	2	1	4	5	6	3	4
UC14	1/7	1/5	1/4	1/4	1/3	1/3	1/2	1/5	1/4	1/4	1/2	1/3	1/4	1	2	3	1/2	1
UC15	1/8	1/6	1/6	1/5	1/4	1/4	1/3	1/6	1/5	1/5	1/3	1/4	1/5	1/2	1	2	1/3	1/2
UC16	1/9	1/7	1/7	1/6	1/5	1/5	1/4	1/7	1/6	1/6	1/4	1/5	1/6	1/3	1/2	1	1/4	1/3
UC17	1/6	1/4	1/4	1/3	1/2	1/3	2	1/4	1/4	1/3	1	1/2	1/3	2	3	4	1	1/2
UC18	1/7	1/5	1/5	1/4	1/3	1/2	1	1/5	1/5	1/4	1/2	1/3	1/4	1	2	3	2	1

Figure 5 Decision Matrix of an Evaluator

Table 6 Weights of Evaluator III and the Results of Consistency

λ_{max}	20,076535
CI	0,122149
RI	1,61
CR	0,075869

# of UC	Weights
1	20,00%
2	10,50%
3	7,20%
4	6,60%
5	4,20%
6	3,00%
7	2,40%
8	9,10%
9	8,60%
10	5,30%
11	3,00%
12	4,40%
13	7,50%
14	1,70%
15	1,20%
16	0,90%
17	2,40%
18	1,90%

Finally, the calculated values of the third evaluator in Figure 5 are: CI value 0.1252149, CR value 0.075869 and RI value 1.61. It was observed as an acceptable matrix, see Table 6.

3.3. Prioritization of Use Cases in Mobile Apps

After performing the analysis of the received questionnaires for Any.Do application, AHP weights and the priority rankings are presented.

Table 7 AHP Weights Results

ID	AHP Weights (%)	Rank
UC1	15,80%	1
UC2	10,50%	2
UC3	7,70%	5
UC4	8,10%	3
UC5	6,90%	6
UC6	7,90%	4
UC7	5,20%	9
UC8	6,40%	7
UC9	6,10%	8
UC10	3,80%	11
UC11	4,00%	10
UC12	3,00%	13
UC13	3,30%	12
UC14	2,80%	14
UC15	1,90%	17
UC16	1,80%	18
UC17	2,20%	16
UC18	2,60%	15

As shown in the Table 7, “Create list” is the most important use case accounting for almost 16% of the total weights followed by “Read/View List with its items” and “Delete list (with its items)” use cases. These use cases are fundamentals of to-do list applications and are considered to be the most important by evaluators. Thus, those use cases should be given priority when building an effective to-do list app. On the other hand, the responders evaluated “Add attachment” and “Share a list item” use cases as the least important ones for to-do list apps.

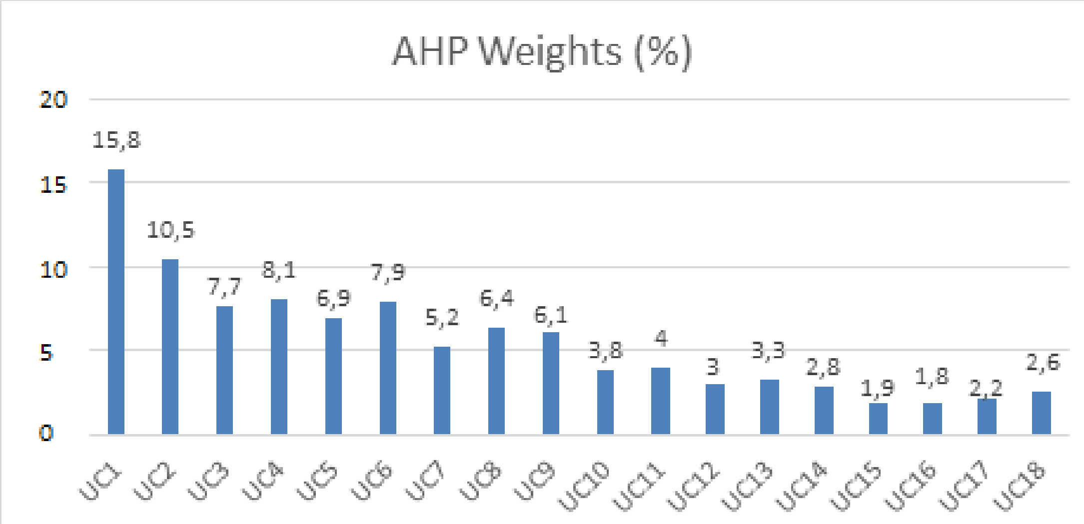

Figure 6 Use Cases Prioritization Chart

According to the answers of the evaluators, all use cases are compared with the others and their values are shown that basic use cases are more important than the others. Also, other features should also not be ignored to attract the user to the application.

As shown in Figure 6, the calculation of AHP method is consistent with the comparison matrices of all double criteria. This is because the consistency ratio (CR) of each matrix is less than 10% [46]. Figure 6 shows the weights of use cases. "Create List" has the most important weight of 15,80%. "Read/View List with its items" is 10.50% and "Update/Rename List" is 7.70%. "Delete List (with its items)" takes the value of 8,10% while "Clear List" is 6,90%. In the least important use cases, "Share a list item" is 1.80%. Another less significant value is "Move an item from one list to another list" 1.90%. This review can be improved by increasing the number of applications and the number of applications for the importance and order of use cases. Figure 6 shows the strong and weak levels of use cases. These results can be developed in the future of productivity applications based on the values of these use cases. Moreover, comparisons can be made by comparing the results with each other.

CHAPTER 4

CONCLUSION

In this study, prioritization of the use of to-do list applications was done by applying AHP method. A systematic method for use case prioritization using AHP was presented. For this reason, 18 use cases are determined, these use cases are the most common in to-do list applications. In these applications, the result indicates that the evaluators view use cases regarding managing lists (e.g. “Create list”, “Read/View List with its items”, “Delete list”) as the most important ones. The first four use cases, which we have been prioritized with AHP methodology, have been found to be successful in the applications with the most interest and the first examples of to-do list applications in Google Play Store. Application developers should consider the observed criteria. The most important issues to be considered when making to-do list applications and application development should be highlighted in these features. These use cases are the common features of the most popular applications. On the other hand, they assess the use cases related to sharing such as “Share list” and “Share a list item” as the least important ones for to-do list apps. These are the lowest ratings available on the Google Play Store. These results can be useful to practitioners to understand how to make the mobile apps fulfilling users’ wants, and preferences. This study also provides insights for researchers by showing that AHP is applicable to prioritize use cases in domain of mobile apps. In this study, the consistency of each responder is checked. Consistency Ratio (CR) was used for this purpose, and if its value is less than or equal to %10 for an evaluator, the responses are considered consistent. The weights of the use cases are calculated based on the evaluations of responders passing the consistency check by using AHP method.

Although the AHP method proved to be applicable to prioritize and select the use cases in the mobile app development. Threads to validity problems are that it suffers from the scalability problem which is due to more of use cases and the use of less use cases is generally more accurate when AHP method is applied, and more comparisons are needed when the number of requirements increases. Further, the approach was only applied in to-do list apps, but it is essential to apply it in different mobile app categories to ascertain its generalizability. Therefore, future research could replicate this study in the development of different mobile app categories.

REFERENCES

[1] I. Malavolta, S. Ruberto, T. Soru, and V. Terragni, “End Users’ Perception of Hybrid Mobile Apps in the Google Play Store,” Proc. - 2015 IEEE 3rd Int. Conf. Mob. Serv. MS 2015, no. August, pp. 25–32, 2015.

[2] W. Martin, F. Sarro, and M. Harman, “Causal impact analysis for app releases in google play,” pp. 435–446, 2016.

[3] A. Möller, S. Diewald, L. Roalter, F. Michahelles, M. Kranz, and S. D. L. R. M. K. A. Möller F. Michahelles, “Update Behavior in App Markets and Security Implications: A Case Study in Google Play,” 3rd Work. Res. large MobileHCI, pp. 5–8, 2012.

[4] N. Viennot, E. Garcia, and J. Nieh, “PlayDrone: A measurement study of google play,” @a@Sigmetrics, pp. 221–233, 2014.

[5] B. Fu, J. Lin, L. Li, C. Faloutsos, J. Hong, and N. Sadeh, “Why people hate your app,” p. 1276, 2013.

[6] N. Chen, J. Lin, S. C. H. Hoi, X. Xiao, and B. Zhang, “AR-miner: mining informative reviews for developers from mobile app marketplace,” pp. 767–778, 2014.

[7] K. Kamvysi, K. Gotzamani, A. Andronikidis, and A. C. Georgiou, “Capturing and prioritizing students’ requirements for course design by embedding Fuzzy-AHP and linear programming in QFD,” Eur. J. Oper. Res., vol. 237, no. 3, pp. 1083–1094, 2014.

[8] S. Nikou and J. Mezei, “Evaluation of mobile services and substantial adoption factors with Analytic Hierarchy Process (AHP),” Telecomm. Policy, vol. 37, no. 10, pp. 915–929, 2013.

[9] M. Sufian, Z. Khan, S. Rehman, and W. Haider Butt, “A Systematic Literature Review: Software Requirements Prioritization Techniques,” 2018 Int. Conf. Front. Inf. Technol., pp. 35–40, 2018.

[10] F. Hujainah, R. B. A. Bakar, M. A. Abdulgabber, and K. Z. Zamli, “Software Requirements Prioritisation: A Systematic Literature Review on Significance, Stakeholders, Techniques and Challenges,” IEEE Access, vol. 6, pp. 71497–71523, 2018.

[11] M. Sadiq, S. Ghafir, and M. Shahid, "An approach for eliciting software requirements and its prioritization using analytic hierarchy process," ARTCom 2009 - Int. Conf. Adv. Recent Technol. Commun. Comput., pp. 790–795, 2009.

[12] M. Pergher and B. Rossi, "Requirements prioritization in software engineering: A systematic mapping study," 2013 3rd Int. Work. Empir. Requir. Eng. Emp. 2013 - Proc., no. August, pp. 40–44, 2013.

[13] R. Wieringa, N. Maiden, N. Mead, and C. Rolland, "Requirements engineering paper classification and evaluation criteria: a proposal and a discussion," Requir. Eng., vol. 11, no. 1, pp. 102–107, 2006.

[14] M. Sadiq, J. Ahmed, M. Asim, A. Qureshi, and R. Suman, "More on elicitation of software requirements and prioritization using AHP," DSDE 2010 - Int. Conf. Data Storage Data Eng., pp. 230–234, 2010.

[15] P. G. Sapna and M. Hrushikesha, "Prioritizing use cases to aid ordering of scenarios," EMS 2009 - UKSim 3rd Eur. Model. Symp. Comput. Model. Simul., pp. 136–141, 2009.

[16] R. Ahmed, D. Musleh, M. Ahmed, and M. El-Attar, "Use case prioritization using fuzzy logic system," Proc. IEEE Int. Conf. Softw. Eng. Serv. Sci. ICSESS, pp. 149–152, 2014.

[17] S. P. Dongoor, "Selecting an appropriate Requirements Based Test Case Prioritization Technique," no. March, 2019.

[18] D. Han and I. Han, "Prioritization and selection of intellectual capital measurement indicators using analytic hierarchy process for the mobile telecommunications industry," Expert Syst. Appl., vol. 26, no. 4, pp. 519–527, 2004.

[19] E. Bagheri, M. Asadi, D. Gasevic, and S. Soltani, "Stratified Analytic Hierarchy Process : Prioritization and Selection of Software Features," pp. 300–315, 2010.

[20] E. N. Maltz, K. E. Murphy, and M. L. Hand, "Decision support for university enrollment management: Implementation and experience," Decis. Support Syst., vol. 44, no. 1, pp. 106–123, 2007.

[21] D. Kundu and D. Samanta, "A novel approach of prioritizing use case scenarios," Proc. - Asia-Pacific Softw. Eng. Conf. APSEC, pp. 542–549, 2007.

[22] C. K. Kwong and H. Bai, “A fuzzy AHP approach to the determination of importance weights of customer requirements in quality function deployment,” J. Intell. Manuf., vol. 13, no. 5, pp. 367–377, 2002.

[23] T.L. Saaty, “{H}ow to make a decision: the analytic hierarchy process,” Eur. J. Oper. Res., vol. 48, no. 1, pp. 9–26, 1990.

[24] G. Işiklar and G. Büyüközkan, “Using a multi-criteria decision making approach to evaluate mobile phone alternatives,” Comput. Stand. Interfaces, vol. 29, no. 2, pp. 265–274, 2007.

[25] F. T. Bozbura and A. Beskese, “Prioritization of organizational capital measurement indicators using fuzzy AHP,” Int. J. Approx. Reason., vol. 44, no. 2, pp. 124–147, 2007.

[26] W. P. PJM Van Laarhoven, “A fuzzy extension of Saaty’s priority theory, Fuzzy Sets and Systems.pdf.” .

[27] M. Chae, J. Kim, H. Kim, and H. Ryu, “Information Quality for Mobile Internet Services: A Theoretical Model with Empirical Validation,” Electron. Mark., vol. 12, no. 1, pp. 38–46, 2002.

[28] H. P. Fu, Y. C. Ho, R. C. Y. Chen, T. H. Chang, and P. H. Chien, “Factors affecting the adoption of electronic marketplaces: A fuzzy AHP analysis,” Int. J. Oper. Prod. Manag., vol. 26, no. 12, pp. 1301–1324, 2006.

[29] S. Perçin, “Use of fuzzy AHP for evaluating the benefits of information-sharing decisions in a supply chain,” J. Enterp. Inf. Manag., vol. 21, no. 3, pp. 263–284, 2008.

[30] O. Kulak and C. Kahraman, “Fuzzy multi-attribute selection among transportation companies using axiomatic design and analytic hierarchy process,” Inf. Sci. (Ny)., vol. 170, no. 2–4, pp. 191–210, 2005.

[31] X. Wang, X. R. Li, F. Zhen, and J. H. Zhang, “How smart is your tourist attraction?: Measuring tourist preferences of smart tourism attractions via a FCEM-AHP and IPA approach,” Tour. Manag., vol. 54, pp. 309–320, 2016.

[32] R. R. Maiti and F. J. Mitropoulos, “Prioritizing Non-Functional Requirements in Agile Software Engineering,” Proc. SouthEast Conf. - ACM SE ’17, pp. 212–214, 2017.

[33] R. K. Shrestha, J. R. R. Alavalapati, and R. S. Kalmbacher, “Exploring the potential for silvopasture adoption in south-central Florida: An application of SWOT-AHP method,” Agric. Syst., vol. 81, no. 3, pp. 185–199, 2004.

[34] Saaty T.L., “Decision making with the analytic hierarchy process,” Int. J. Serv. Sci., vol. 1, no. 1, p. 83, 2008.

[35] N. Afzal and M. Sadim, “A p p l y i n g A n a l y t i c H i e r a r c h y P r o c e s s f o r t h e S e l e c t i o n o f t h e R e q u i r e m e n t s o f I n s t i t u t e E x a m i n a t i o n S y s t e m,” pp. 94–102.

[36] N. Garg, M. Sadiq, and P. Agarwal, GOASREP: Goal oriented approach for software requirements elicitation and prioritization using analytic hierarchy process, vol. 516. 2017.

[37] C. F. Lai, P. S. Chiu, Y. M. Huang, T. S. Chen, and T. C. Huang, “An evaluation model for digital libraries’ user interfaces using fuzzy AHP,” Electron. Libr., vol. 32, no. 1, pp. 83–95, 2014.

[38] I. Basak and T. Saaty, “Gruop Decision Making usin AHP,” Math. Comput. Model., vol. 17, no. 415, pp. 101–109, 1993.

[39] L.-H. M. J. Barcenas G. R., “Saaty’s Analytic Hierarchies Method for KnowledgeOrganization in Decision Making,” 2013.

[40] S. Unutmaz, “Factors Affecting Job Satisfaction Among Employees in a Public Institution,” Middle East Tech. Univ., no. December, pp. 1–129, 2014.

[41] R. N. Jones, “IPCC-WGII-AR5-02. Foundations for Decision Making,” Contrib. Work. Gr. II to Fifth Assess. Rep. Intergov. Panel Clim. Chang., no. December, pp. 195–228, 2014.

[42] M. F. B. Salomon, C. H. P. Mello, and E. G. Salgado, “Prioritization of product-service business model elements at aerospace industry using analytical hierarchy process,” Acta Sci. Technol., vol. 41, no. 1, p. 37934, 2019.

[43] C.-P. L. S.-J. S. R.-C. T. K.-H. Lee, “An AHP-Based Weighted Analysis of Network Knowledge Management,” Turkish Online J. Educ. Technol. - TOJET, vol. 10, no. 4, pp. 52–59, 2011.

[44] J. Franek and A. Kresta, “Judgment Scales and Consistency Measure in AHP,” Procedia Econ. Financ., vol. 12, no. March, pp. 164–173, 2014.

[45] T. Saaty and L. G. Vergas, Models , Methods , Concepts & Applications of the Analytic Hierarchy Process Second Edition, vol. 175. 1996.

[46] T. L. Saaty, “How to Make a Decision: Advice for Scientists,” European Journal of Operational Research, vol. 48, no. 48. pp. 9–26, 1990.