SOFTWARE SIZE ESTIMATION PERFORMANCE OF SMALL AND MIDDLE SIZE FIRMS IN TURKEY

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

BY

ERDEM ÇOLAK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN THE PROGRAM OF
SCIENCE AND TECHNOLOGY POLICY STUDIES

AUGUST 2010

Approval of the Graduate School of Social Sciences

Prof.Dr.Meliha Altunışık
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Assoc.Prof.Dr.Erkan Erdil
Head of Department

This is to certify that we have read this thesis 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.Erol Taymaz
Supervisor

Examining Committee Members (first name belongs to the chairperson of the jury and the second name belongs to supervisor)

Assoc.Prof.Dr.Erkan Erdil	(METU, ECON)	
Prof.Dr.Erol Taymaz	(METU, ECON)	
Dr.Ekrem Kalkan	(Competition Authority)	

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: Erdem Çolak

Signature :

ABSTRACT

SOFTWARE SIZE ESTIMATION PERFORMANCE OF
SMALL AND MIDDLE SIZE FIRMS IN TURKEY

Çolak, Erdem
MSC. Department of Science and Technology Policy Studies
Supervisor : Prof. Dr. Erol Taymaz

August 2010, 171 pages

Software cost estimation is essential for software companies to be more competitive and more profitable. The objective of this thesis is to study current software size estimation practices adopted by Turkish software companies, to identify best practices, and to suggest appropriate methods that can help companies to reduce errors in their software size estimations.

Keywords: Software Size, Estimation

ÖZ

TÜRKİYE'DE KÜÇÜK VE ORTA BÜYÜKLÜKTEKİ FİRMALARININ YAZILIM BÜYÜKLÜĞÜ KESTİRİM PERFORMANSI

Çolak, Erdem
Lisansüstü, Bilim ve Teknoloji Politikaları Araştırmaları Bölümü
Tez Yöneticisi: Prof. Dr. Erol Taymaz

Ocak 2010, 171 sayfa

Yazılım maliyet tahmini, yazılım firmalarının daha rekabetçi ve karlı olmaları için önemlidir. Bu tezin amacı Türk yazılım firmalarında başvurulan maliyet tahmin uygulamalarını araştırmak, en iyi uygulamaları belirlemek ve firmalara uygun yöntemleri önererek maliyet tahminlerindeki hatalarını azaltmakta yardımcı olmaktır.

Anahtar Kelimeler: Yazılım Maliyeti, Tahmin

To My Wife and My Son

ACKNOWLEDGMENTS

The author wishes to express his deepest gratitude to his supervisor Prof. Dr. Erol Taymaz for his guidance, advice, criticism, encouragements and insight throughout the research.

The author would like to thank to Mr. Sabri Gökmenler, Manager of IT Architecture and Security Division of Is Bank for his suggestions and support.

The author would also like to thank to Mrs. Meriç Merih Aykol for her excellent ideas and encouraging me to complete this program.

The assistance of Mr. Tankut Tekeli and Mr. Hasan Çağlar are gratefully acknowledged.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Çolak, Erdem
Nationality: Turkish (TC)
Date and Place of Birth: 11 May 1975, Kahramanmaraş
Marital Status: Married
Phone: +90 212 3659890
Fax: +90 212 3659898
Email: erdem.colak@isbank.com.tr

EDUCATION

Degree	**Institution**	**Year of Graduation**
BS	METU Electrical and Electronics Engineering	1999
High School	Atatürk High School, İzmir	1992

WORK EXPERIENCE

Year	**Place**	**Enrollment**
2004- Present	SoftTech A.Ş.	Manager
1999-2004	Türkiye İş Bankası A.Ş.	Software Profession

FOREIGN LANGUAGES

Advanced English

HOBBIES

Model Planes, Playing Guitar, Watching Documentaries

ABBREVIATIONS

Abbreviation	Compound Term
API	Application Programming Interface
CMMI	Capability Maturity Model Integrated
COCOMO	The Constructive Cost Model
COTS	Commercially Of The Shelf Software
DI	Degree of Influence
EEA	Effort Estimation Accuracy
EF	Environmental Factor
EU	European Union
FP	Function Point
FPA	Function Point Analysis
GNP	Gross National Product
IAT	Informatics Associations of Turkey
ID	Identification
IFPUC	International Function Point Users Group
IP	Internet Protocol
IT	Information Technologies
KLOC	Kilo Lines of Code
LOC	Lines of Code
OECD	Organization for Economic Co-operation and Development
SEER-SEM	Software Evaluation and Estimation of Resources – Software Estimating Model
SLIM	Software Lifecycle Management (QSM Associates tool)
TBD	Türkiye Bilişim Derneği
TCF	Technical Complexity Factor
TCP	Transport Control Protocol
TDI	Total Degree of Influence
TUİK	Türkiye İstatistik Kurumu
UAW	Unadjusted Actor Weight

UCP	Use Case Points
UFP	Unjustified Function Points
USA	United States of America
UUCP	Unadjusted Use Case Points
UUCW	Unadjusted Use Case Weight
WBS	Work Breakdown Structure
XP	Extreme Programming

CHAPTER 1 INTRODUCTION

The formal dictionary definitions of the word "estimate" are: 1) A judgment or opinion about the value or quality of somebody or something. 2) A judgment about the levels or quantity of something (Oxford Advanced Learner's Dictionary, 2005).In the software sector, the word "estimation" means the preliminary calculation of cost of the software project. Early calculation of cost is important in all business sectors. However, it is more important in the software sector since the complex and uncertain nature of the software development process frequently results in cost and schedule overruns.

Cost estimation for software projects is more important in developing countries like Turkey. Minimizing the error rate of the cost estimate becomes vital for small and medium sized enterprises when it comes to tenders for new projects. The difference between estimated and actual costs of a software project might mean bankruptcy of a software firm which has limited resources.

The objective of this thesis is to analyze current estimation practices in the Turkish software sector, to identify the best practices among them, to suggest appropriate methods while taking into consideration of special conditions of the small and medium sized enterprises, and to provide a basis for further studies on software cost estimation in Turkish software sector.

The method used to determine the current state of the software sector was the survey method. The survey was composed of 46 questions divided into seven groups on organizational structure, software life-cycle process maturity of the organization, estimation details of the sample project, software product constraints, project team experience, production environment constraints, and project properties. The survey was designed to determine software cost items, organization, project and software product characteristics, maturity level of the software life-cycle processes, and the estimation practices of the organization.

The survey was conducted through face to face interviews and by sending the electronic survey forms to individuals contacted with the help of the social networks. The respondents were employers, managers, directors, program managers, project managers, and team leaders who have project management experience in their professional careers. The respondents are knowledgeable about project characteristics, project cost details, team structure and experience, and organizational structure.

The thesis is organized as follows; the subject, objective, and research method are summarized in the first section. The second section includes an assessment of the Turkish software industry based on the statistics provided by official sources and non-governmental organizations. The third section provides a summary of estimation methods. In this section, estimation methods are logically classified into four groups which are methods based on expert judgment, decomposition and recomposition methods, algorithmic methods and proxy methods. The fourth section presents the analysis and interpretation of the survey results. The fifth section summarizes the main findings and conclusions of the thesis.

CHAPTER 2 SOFTWARE INDUSTRY IN TURKEY

2.1 Software Market Size

It is estimated that the market size of the information technologies is about 3 billion dollars in Turkey. Figure 1 shows the market size of hardware, commercial off the shelf software (COTS) and information technology (IT) services.

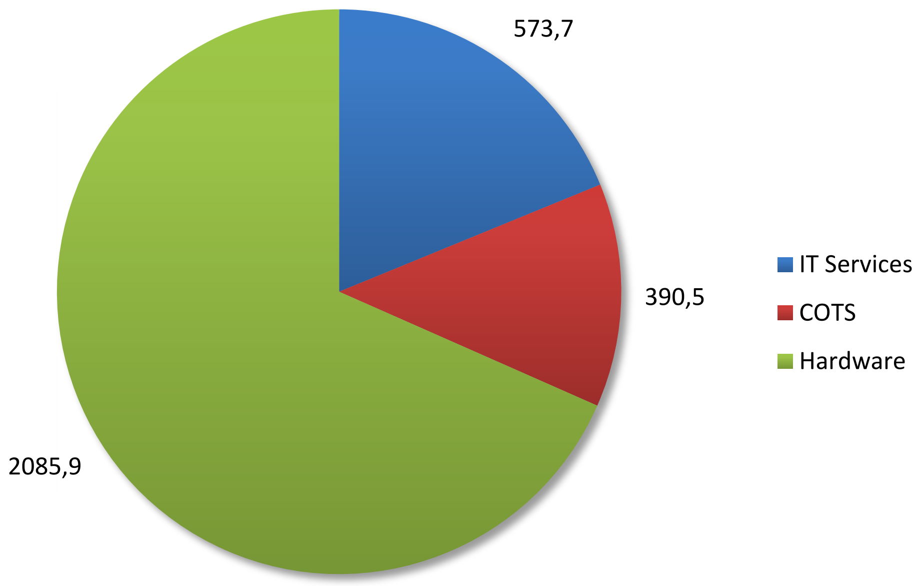

Figure 1 Information Technologies Market in Turkey, 2005 (Million $)

(Source: IDC)

The hardware market comprises about 68% of the Turkish IT market whereas it is 40% in OECD countries.

Table 1 Comparison of the Turkish IT Market with International Markets with Respect to IT Sectors

	West Europe	East Europe	USA	Japan	Other Countries	World	Turkey
IT Hardware	42,8	71,1	34,4	59,3	71,7	49,2	68,4
Software	19,6	11,1	23,4	11,9	8,7	17,9	12,7
IT Services	37,6	17,8	42,2	28,8	19,6	32,9	18,9

(Source: İnterpromedya, 2005)

It can be seen from the Table 1 that, the ratio of the hardware component of the Turkish IT market is above the world averages. In addition, Table 1 shows that the IT market of U.S. and Europe is mainly composed of IT services. As the extent of integration of IT technologies with business increases, the productivity and competitiveness of the whole market increases. In this respect, IT services and Software have the same importance in comparison with IT hardware. Not including Japan, developed countries in the Table 1 prove this view. The current hardware weighted state of the Turkish Market indicates that the level of integration of IT Technologies has not reached the desired level of integration to support productivity and competitiveness. (DPT, 2007, pp. 2-3)

2.2 Software Sector Profile

The size of the IT market in Turkey is not accurately known but can be estimated. However, IT market production data is not available and the business segments that demand IT products are not known. Yet, some information about the firms operating in Technology Development Zones which are mostly known as Teknoparks and the business segment they serve can be given.

The number of Teknoparks in Turkey has increased gradually. The 21st Teknopark was established in September 2005. Among the 21 Teknoparks, 10 are active. 67%

of the firms in Teknoparks are operating in the software market. 382 firms are operating in 10 Teknoparks and 252 of them are operating in the IT market. (DPT, 2007, pp. 10-11)

Table 2 Business Sectors of the Firms Operating in Technology Development Zones

	Number of Firms	IT	Defense	Electronics	Telecommunication	Advanced Materials	Medicine	Other
ODTÜ Teknokent	135	58	28	14	3	1	1	30
HACETTEPE	25	14	-	4	-	-	1	6
GOSB	13	12	-	-	-	-	-	1
TÜBİTAK-MAM	26	18	-	-	-	8	-	-
İTÜ Arı Teknokent	23	23	-	3	3	-	-	3
İZMİR	13	11	-	1	-	1	-	-
Bilkent CyberPark	108	94	15	28	18	2	3	39
ESKİŞEHİR	6	3	-	2	-	1	-	1
KONYA	28	15	-	1	-	-	2	10
BATI AKDENİZ	5	4	-	1	-	-	-	-
Total	**382**	**252**	**43**	**54**	**24**	**13**	**7**	**90**

(Source: DPT - Department of Industry and Commerce October 2005)

According to the Informatics Association of Turkey (TBD), the number of IT firms operating in Turkey is unofficially estimated at around 7,000 with 5,500 of them operating in a steady manner. It is predicted that about 10,000 firms and internal organizations have been founded for IT needs. Around 2,500 of the 10,000 firms and internal organizations are predicted to be in IT software or IT software related institutions. The most remarkable point is that half of the IT firms go bankrupt in economic crises and new firms are founded to replace the bankrupt firms.

According to Turkish Statistical Institute (TÜİK) statistics, the average costs of employing an employee in different sectors are shown in Table 3

Table 3 Average Cost to Employ an Employee in Different Sectors

Sector	**Average Cost of Employing an Employee (US Dollars)**
Information Technologies	3.500
Agriculture	40.000
Tourism	55.000
Industry	90.000

(Source: Kavrakoğlu, 2003, p. 15)

The average value added by an employee in different sectors is given in Table 4.

Table 4 Average Values Added by an Employee in Different Sectors

Sector	**Average Value Added by an Employee (US Dollars)**
Information Technologies	30.000
Services	16.000
Industry	8.000
Agriculture	4.000

(Source: Kavrakoğlu, 2003, p. 15)

Despite the opportunities in the IT sector, the percentage of the IT sector with respect to GNP is lower in Turkey than that in the EU and USA.

Table 5 Percentage of IT Sector with Respect to GNP of Turkey and Developed Countries

Country	Percentage of IT Sector with respect to GNP
Turkey	%0,8
EU Countries	%2,5
USA	%4,5

(Source: Kavrakoğlu, 2003, p. 17)

Family owned enterprises are the dominant structures in small and middle sized enterprise sectors in Turkey. Even in holding companies, the level of institutionalization is low. In these circumstances, the realization of concepts like “Informatics”, “Knowledge Society” and “Software” is difficult. The half of the Turkish economy is thought be off the out of record and this half is against computerization and recording. This situation is the one of most important obstacles to the advancement of Turkish IT sector.

The Turkish software sector is small and negligible compared to the world scale. Nevertheless, there exist success stories like Logo, Netsis, Cybersoft, Havelsan, Milsoft, Meteksan, Telenity, Coretech, SFS, Veripark, and SoftTech, particularly if the conditions they came from are considered. Although these firms prove that Turkish type successes can be achieved if the correct strategies are applied, most Turkish software firms are weak in the areas of strategic planning and institutionalization.

The best example of the changes in the software sector can be seen in the banks. The banking sector which had previously insisted on internal software development has started to obtain complex software from outside. The evolutionary change seen in banking sector has not yet been seen in other sectors. . The insistence on developing software internally along with incorrect decisions about outsourcing cause firms to lose money, managers to lose their jobs and may even cause firms to go bankrupt. In the past, the unsuccessful cases gave rise to big businesses heading for international software companies. However, international software products are large and extensive for the Turkish firms. Therefore, standing in front of the Turkish software industry is a software market which is mostly unsaturated and comprised largely of small and middle sized enterprises.

According to the Software Industrialists Association, the Turkish IT market reached 3.5 billion dollars and 300 million dollars of this market is software. The total sales of the local software firms are about 60 million dollars if the sales of the foreign software firms are excluded.

While most of the software firms in Turkey produce one type of software product, a few of them produce more than one type of software product. Except for a few examples, the Turkish software sector has not managed to create trademarks. This state of the Turkish software sector has resulted in heading towards foreign software firms.

Another topic of discussion in the software sector is the “project or product?” discussion. Some authorities suggest heading towards packaged software because of economies of scale whereas some suggest heading towards producing a solution since it is difficult to compete with foreign software packages. According to software solution supporters, the change of Turkish software firms in software products is low since the margin of profit is low in software products. Yet, financing software projects is easier since the down payments are usually rendered at the early stages of the software projects.

Turkish software firms seem to have more advantage in small and middle sized enterprises because of the common language and shared business culture, whereas foreign software firms seem to have more chance because of alignment with standards, quality and an inclination for foreign product preference. The small size of the Turkish software firms turns in to an advantage for the foreign software firms. The best example of disadvantage of Turkish software firms can be seen in the Strategy for Information Society. In the Strategy for Information Society, software did not attract much attention and the benefits of the local software firms were not discussed. Another example of this disadvantage can be seen in E-Transformation (E-Dönüşüm) project. In this project, local software firms were not represented. (Alican, 2006, pp. 91-99)

CHAPTER 3 SOFTWARE SIZE ESTIMATION METHODS

3.1 Methods Based on Experience and Expert Judgment

Estimation methods on the basis of experience and expert judgment refer to the consideration of expert opinions and experience. Estimation methods which fall into this group are very common estimation approaches in IT projects. Estimation can be performed individually by one or several experts or by estimation groups. The estimation methods based on expert judgment and experience are grouped into three parts; Individual Expert Judgment, Estimation by Analogy, and Delphi Method. Individual Expert Judgment and Estimation by Analogy are the examples of estimation methods performed individually whereas the Delphi Method is an example of an estimation method performed by an estimation group.

3.1.1 Individual Expert Judgment

Expert judgment is an estimation strategy conducted by person who is an expert on the task being estimated. In the software branch of business, the expert is a person who has experience with the technology or concerning the development practice. The remarkable part of the expert judgment process is non-explicit, non-recoverable and intuitive compared to most estimation processes which have both intuitive and explicit reasoning elements. However, the individual expert judgment process does not have to be completely intuitive or informal. Both estimation practices which are utterly intuitive and estimation practices which are supported by guidelines, past project data sets, process definitions and control lists i.e. “structured expert estimation” fall in to this category (Jørgensen, 2002, p. 2).

In the software industry, expert judgment is by far the dominant estimation method in practice. (Jørgensen, 2002, pp. 2-3). Hihn and Habib-Agahi reported that 83% of the estimators uses “informal analogy”, 4% “formal analogy”, 6% “rules of thumb”, and 7% “models” as their primary estimation method for software development effort estimation at NASA Jet Repulsion Laboratory (Hihn & Habib-Agahi, 1991, pp. 276-287). The survey conducted in New Zealand showed that 86% of the software organizations use expert judgment as an estimation method (Paynter, 1996, pp. 150-159). Kitchenham and her colleagues found that 72% of the project estimates of a software development company were based on “expert judgment” (Kitchenham, Pfleeger, McColl, & Eagan, 2002).

3.1.2 Estimation by Analogy

Estimation by analogy is sometimes described as a systemic form of the expert judgment in a way that experts express their opinion using analogous situations. (Shepperd, Schofield, & Kitchenham, 1996, pp. 170-178) The idea behind the estimation by analogy method is to find a completed and very similar project; and use the size and the effort data to use in estimating the new project. If the new project is different by more than ±25%, another completed project should be found or an alternative estimation method should be applied. (Laird & Brennan, 2006, p. 88). Estimating project effort by analogy involves following steps (Walkerden & Jeffery, 1999, pp. 135-158):

1) Measuring or estimating project metrics of the target (new) project.
2) Selecting a source (completed) project from asset library or project repository of the organization which is very similar to the target project.
3) Assuming the effort value of the source project as the initial value of the target project.
4) Comparing the project metrics of source and target project to calculate multiplier factor for each project metrics.
5) Estimate the effort of target project using multiplier factors.

Steve McConnell illustrates the estimation by analogy method using two imaginary projects AccSellerator 1.0 and Triad 1.0. AccSellerator 1.0 is a completed project composed of 5 subsystems:

Table 6 Size of the Imaginary Project AccSellerator 1.0

Database	5.000 LOC
User interface	14.000 LOC
Graphs and reports	9.000 LOC
Foundation classes	4.500 LOC
Business rules	11.000 LOC
TOTAL	**43.500 LOC**

(Source: McConnell, 2006, p. 128)

In the next step of his illustration, McConnell gives the metrics of the subsystems of the projects:

Table 7 Metrics of the Imaginary Project AccSellerator 1.0

Database	10 Tables
User interface	14 Web pages
Graphs and reports	9 Graphs + 8 reports
Foundation classes	15 Classes
Business rules	-

(Source: McConnell, 2006, p. 129)

Table 8 Metrics of the Imaginary Project Triad 1.0

Database	14 Tables
User interface	19 Web pages
Graphs and reports	14 Graphs + 16 reports
Foundation classes	15 Classes
Business rules	-

(Source: McConnell, 2006, p. 129)

Table 9 Calculation of Multiplication Factors of Each Subsystem

Subsystem	**Actual Size of AccSellerator 1.0**	**Estimated Size of Triat 1.0**	**Multiplication Factor**
Database	10 Tables	14 Tables	1,4
User interface	14 Web pages	19 Web pages	1,4
Graphs and reports	9 Graphs + 8 reports	14 Graphs + 16 reports	1,7
Foundation classes	15 Classes	15 Classes	1,0
Business rules	-	-	1,5

(Source: McConnell, 2006, p. 129)

McConnell estimates the size of each subsystem of Triad 1.0 by multiplying the actual size of each subsystem of AccSellerator 1.0 with a corresponding multiplication factor:

Table 10 Estimated Size of Triad 1.0

Subsystem	Code Size of AccSellerator 1.0	Multiplication Factor	Estimated Code Size of Triad 1.0
Database	5.000 LOC	1,4	7.000 LOC
User interface	14.000 LOC	1,4	19.600 LOC
Graphs and reports	9.000 LOC	1,7	15.300 LOC
Foundation classes	4.500 LOC	1	4.500 LOC
Business rules	11.000 LOC	1,5	16.500 LOC
TOTAL	**43.500 LOC**		**62.900 LOC**

(Source: McConnell, 2006, p. 130)

Knowing the estimated code size of the imaginary project Triad 1.0, McConnell uses (Equation 1) for calculating the size ratio of the two projects:

$$\text{Size Ratio} = \frac{\text{Estimated Code Size New Project}}{\text{Code size of Completed Project}} \qquad \text{(Equation 1)}$$

Then, the size ratio of Triad 1.0 and AccSellerator 1.0 is computed as:

$$\text{Size Ratio} = \frac{62.900}{43.500} = 1{,}45$$

Finally, McConnell calculates the estimated effort of Triad 1.0:

Estimated Effort = Size Ratio x Actual Effort of Completed Project (Equation 2)

Estimated Effort = 1.45 x 30 Staff Months

Estimated Effort = **44 Staff Months** (McConnell, 2006, pp. 128-132)

3.1.3 Delphi and Wide Band Delphi

The Delphi Method is developed in the U.S. Air Force sponsored RAND Corporation by conducting a series of experiments in 1950s. The objective of the study was to develop a method in order to obtain the closest to a true answer out of the expert group. The Delphi method is based on the assumption that, the probability that the relevant information exists in the minds of experts in a group is higher than the probability of the relevant information found in a mind of individual expert. On the other hand, the probability of misinformation being aggregated in a group judgment equally increases at the same time.

“Committees, councils, panels, commissions, juries, boards, the voting public, legislatures... the list is long, and illustrates the extent to which the device of the pooling many minds has permeated society.” (Dalkey, Norman C.;, 1969, p. 6)

The traditional way of extracting opinions is face to face discussions. However, significant number of studies done by physiologists show that face to face discussions run the risk of influence of dominant contributors and noise. Noise is not the level of sound. By noise what is meant is the “communication” between contributors which is related to their personal interests, rather than the problem. Herein, The Delphi Method defines a procedure to remove political bias, the pressure of dominant contributors and noise. It outlines a technique for extracting reliable opinions from expert groups.

In the case where the judgment is a numerical estimation e.g. the year that a certain technological achievement occurred, the population of the world in 2050 or staff months required to develop a specific software-, the median of the answers of the indistinguishable experts (M), is close to the true answer (T) at least one half of the group answers.

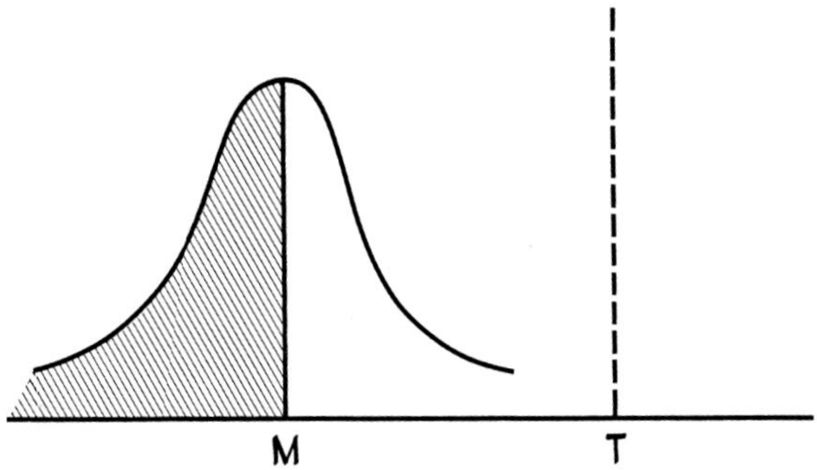

Figure 2 Worst case: Median better than half

(Source: Dalkey, Norman C.;, 1969, p. 8)

In the normal situations, M is closer to T more than half of the expert group.

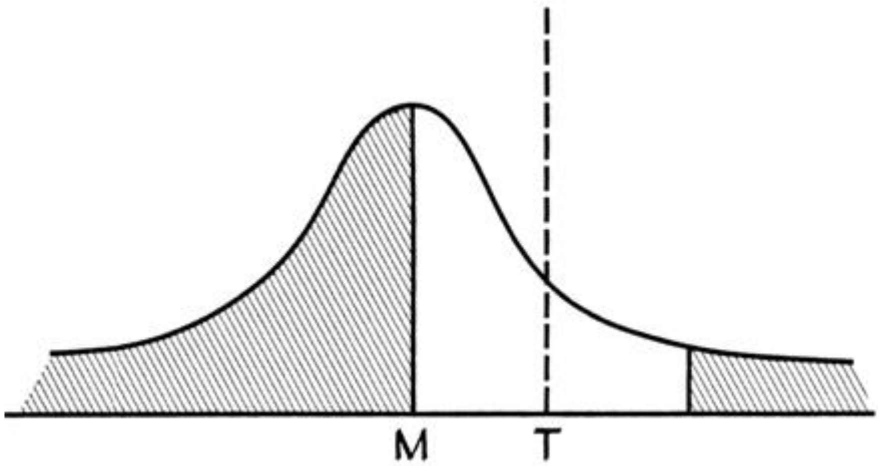

Figure 3 Normal case: Median better than more than half

(Source: Dalkey, Norman C.;, 1969, p. 9)

One of the interesting results of the experiments conducted by the RAND Corporation is that the error of the group response decreased and the reliability of the group opinions increased as the group size increased. The largest group in the experiment was a group composed of 29 experts. (Dalkey, Norman C.;, 1969, pp. 7-14)

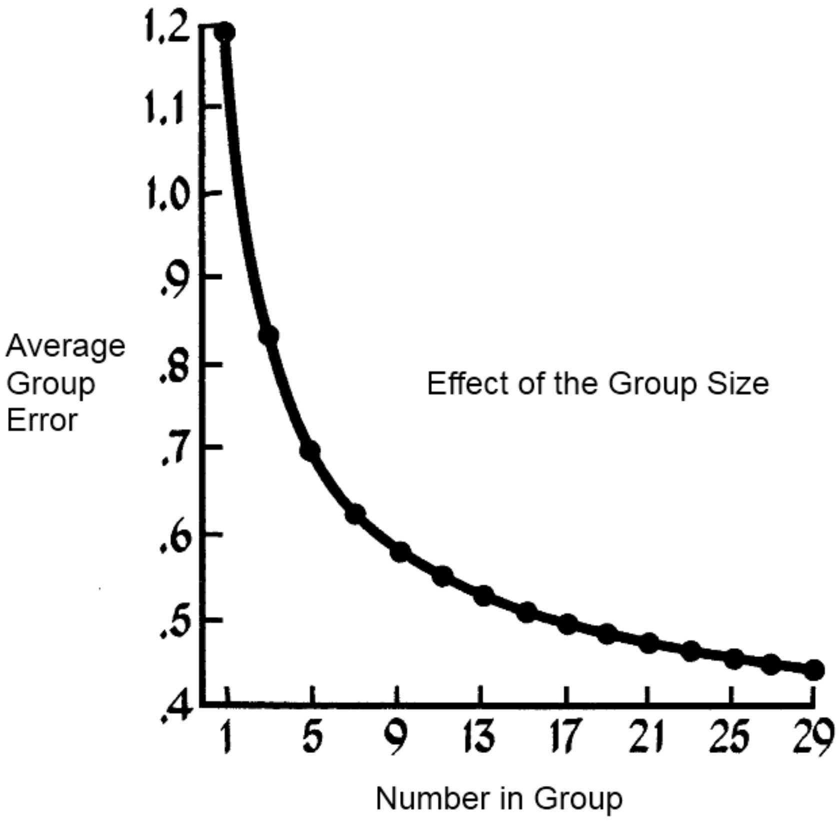

Figure 4 Effect of the Group Size

(Source: Dalkey, Norman C.;, 1969, p. 11)

The Delphi Method has three main characteristics:

1. Autonomy
2. Controlled feedback
3. Statistical group response.

Autonomy is provided by formal communication channels such as online computer programs or with the help of questionnaires to reduce the effect of dominant contributors. Controlled feedback is the sequence of rounds where the results of the previous round are evaluated. Controlled feedback is the method for reducing noise. Statistical group response is used for reducing the pressure of conformity and provides a formal way to assure that every opinion is represented in the final result. (Dalkey, Norman C.;, 1969, p. 16)

The steps of the Delphi Methods are the following: (Rowe & Wright, 2001, p. 126)

a) Moderator defines elements to be estimated.
b) Moderator determines the expert group. The suggested group size is between 5 and 20 experts.
c) Each expert prepares an estimation.
d) Moderator presents estimations anonymously. Moderator presents estimation result in the iteration as statistical summary of group response, mean, median of each estimation or average of all estimations.
e) Moderator organizes next iteration. (Return to step c))
f) Moderator ends the iteration cycle when the estimations of the group become stable or converge to a single point estimation.

The Wide Band Delphi Method is same as the Delphi Method. The only difference is the group discussion in the Wide Band Delphi Method. The steps of the Wide Band Delphi are the following: (McConnell, 2006, p. 151) (Hennessy, 2004)

a) Delphi moderator presents each estimator with an estimation form.
b) Delphi moderator conducts group discussion
c) Estimators prepare initial estimates.
d) Each estimator gives their estimations individually and anonymously.
e) Delphi moderator prepares a summary of estimations. Estimators can compare their estimates with the others' estimates.
f) Estimators discuss reasons of the variations in their estimations.
g) Estimators vote anonymously on whether they accept the average estimate or not. If any of the estimators votes "No", group returns to step c).
h) The result is either single point estimation which is expected result or range of estimations created through group discussion.

The first application of the Delphi method was an application of "expert opinion to the selection, from the point of view of a Soviet strategic planner, of an optimal U. S.

industrial target system and to the estimation of the number of A-bombs required to reduce the munitions output by a prescribed amount." (Dalkey & Helmer, 1963, p. 458) The alternative method of handling this problem involves costly data collection, and use of expensive and laborious computer power. Even if the alternative method is applied, there still exist subjective parameters which can directly affect results. When the information is difficult and expensive to obtain or subjectivity is involved in the parameters of the alternative estimation methods, the Delphi Method is a good choice for obtaining more accurate estimation from expert groups (Linstone & Turoff, 2002, p. 10)

3.2 Decomposition and Re-Composition Methods

Decomposition and re-composition methods can be compared to deduction and induction methods in science. Decomposition which is widely known as the Top-Down Approach and Re-Composition which is widely known as the Bottom-Up Approach are estimation techniques opposite of each other. Both the Top-Down and the Bottom-Up Approaches can be applied to project tasks or systems composed of subcomponents.

3.2.1 Top-Down Approach

The Top-Down approach is an estimation method in which software project effort is derived from global properties of the project. When the total effort to complete the project is estimated, the effort is distributed over the sub systems to be developed or the sub activities of the software project to be carried out. (Boehm, 1981, p. 337)

The Top-Down Approach is a subsidiary technique. Once the overall estimate for the project is established, the estimated effort is divided through the layers of the work breakdown structure. Top-down budgets are often used by organizations that complete IT projects for other companies. (Güzel, 2009, pp. 24-25)

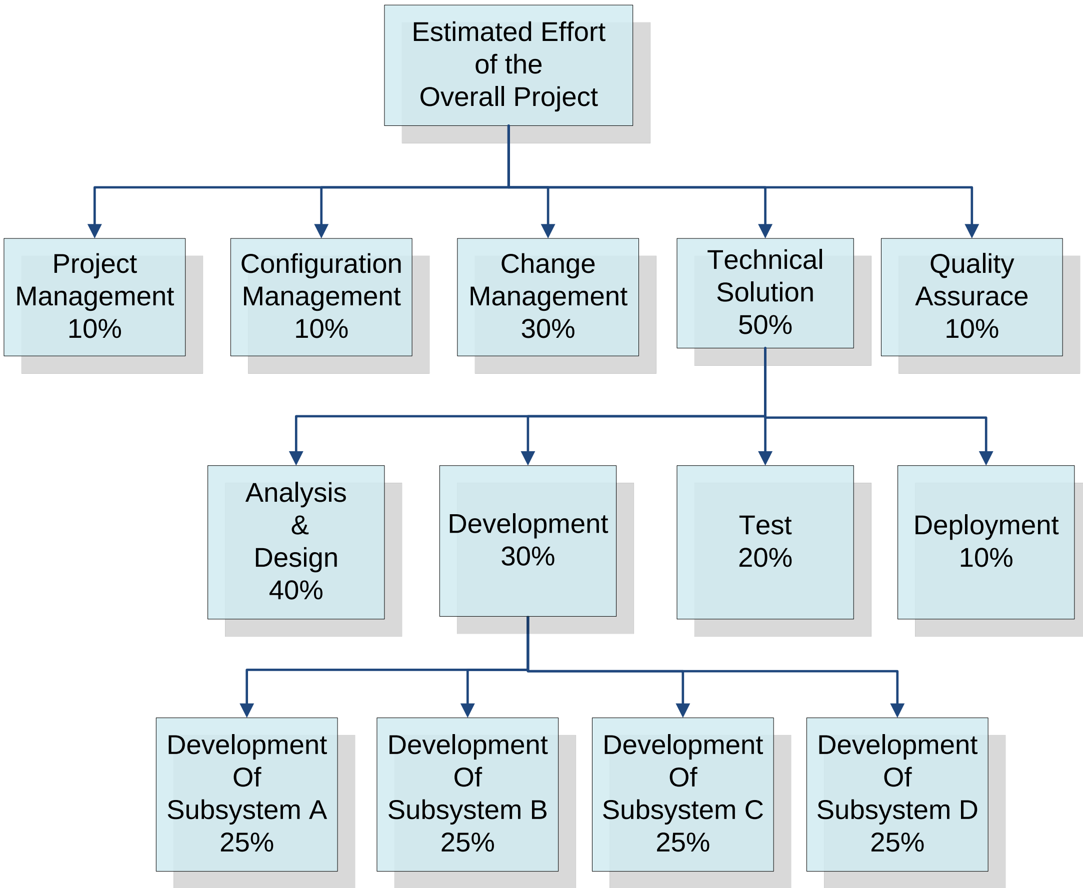

Figure 5 Example of Top-Down Approach

The main disadvantage of the Top-Down Approach is that it does not to identify cost driving low level technical problems and components to be developed. Furthermore, it does not provide justification for effort distribution over sub components of project. (Wolverton, 1974, pp. 615-636)

3.2.2 Bottom-Up Approach

In the Bottom-Up Approach, the effort required to develop each software component is estimated by the individual who is responsible for development of that component. Estimations are added up in order to obtain the total effort to complete the project. The Bottom-Up approach is the opposite of the Top-Down Approach. While the Bottom-Up approach focuses on the effort estimation of each software component to be developed, it ignores system level efforts such as configuration management, integration, project management etc. As a result of this characteristic, the Bottom-Up Approach results in an underestimated total effort of the project. On the other hand, having the particular job to be estimated by the individual who is responsible of that job has two advantages:

1. Each of the component estimations is based on a detailed understanding of job to be done.
2. Estimations are more reliable since they are done by the individuals who are responsible from development of those components.

The best way to ensure system level activities (integration, configuration management, etc.) is to organize project activities in the work breakdown structure (WBS). WBS includes the hierarchical structure of the jobs to be completed in the project. However, effort estimation of the system level activities should be delayed until the completion of the estimation of every component in the system. For example, estimation of the effort required for integration activities before estimation of development effort of each component in the system would probably result in error in the total project effort. (Boehm, 1981, pp. 338-339)

3.3 Algorithmic Methods

The algorithmic models are mathematical models derived from industrial data obtained from a certain environment. The simplest ones can be used manually whereas complex models are embedded in software tools.
Algorithmic models have an intermediate step to estimate size in the units of KLOC or FP, and then calculate effort using size estimation. Using effort estimation, project schedule and staffing are estimated. (Laird & Brennan, 2006, p. 103)

3.3.1 Manual Models

The typical algorithmic model is in the form of equation below:

$$\text{Effort} = A + B \times (\text{Size})^C \qquad \text{(Equation 3)}$$

The parameters *A*, *B* and *C* in the (Equation 3) are constants determined from empirical data. The factor *B* is known as productivity factor. Smaller *B* implies higher productivity in the project. Another important factor *C* is known as the scaling factor. The factor *C*, having values greater than 1 implies dis-economies of scale whereas

C having values smaller than 1 implies economies of the scale. Size variable in the (Equation 3) is either KLOC or FP. (Laird & Brennan, 2006, p. 103)

There are many different (Equation 3) type KLOC equations calibrated to project data sets. For example: (McGibbon, 1997):

Table 11 Different KLOC Equations

Equation Name	**Equation**	
Walston-Felix	Effort = $5.2 \times (KLOC)^{0.91}$	(Equation 4)
Bailey-Basili	Effort = $5.5 + 0.73 \times (KLOC)^{1.16}$	(Equation 5)
Boehm Simple	Effort = $3.2 \times (KLOC)^{1.05}$	(Equation 6)
Doty	Effort = $5.288 \times (KLOC)^{1.047}$	(Equation 7)

(Source: Laird & Brennan, 2006, p. 103)

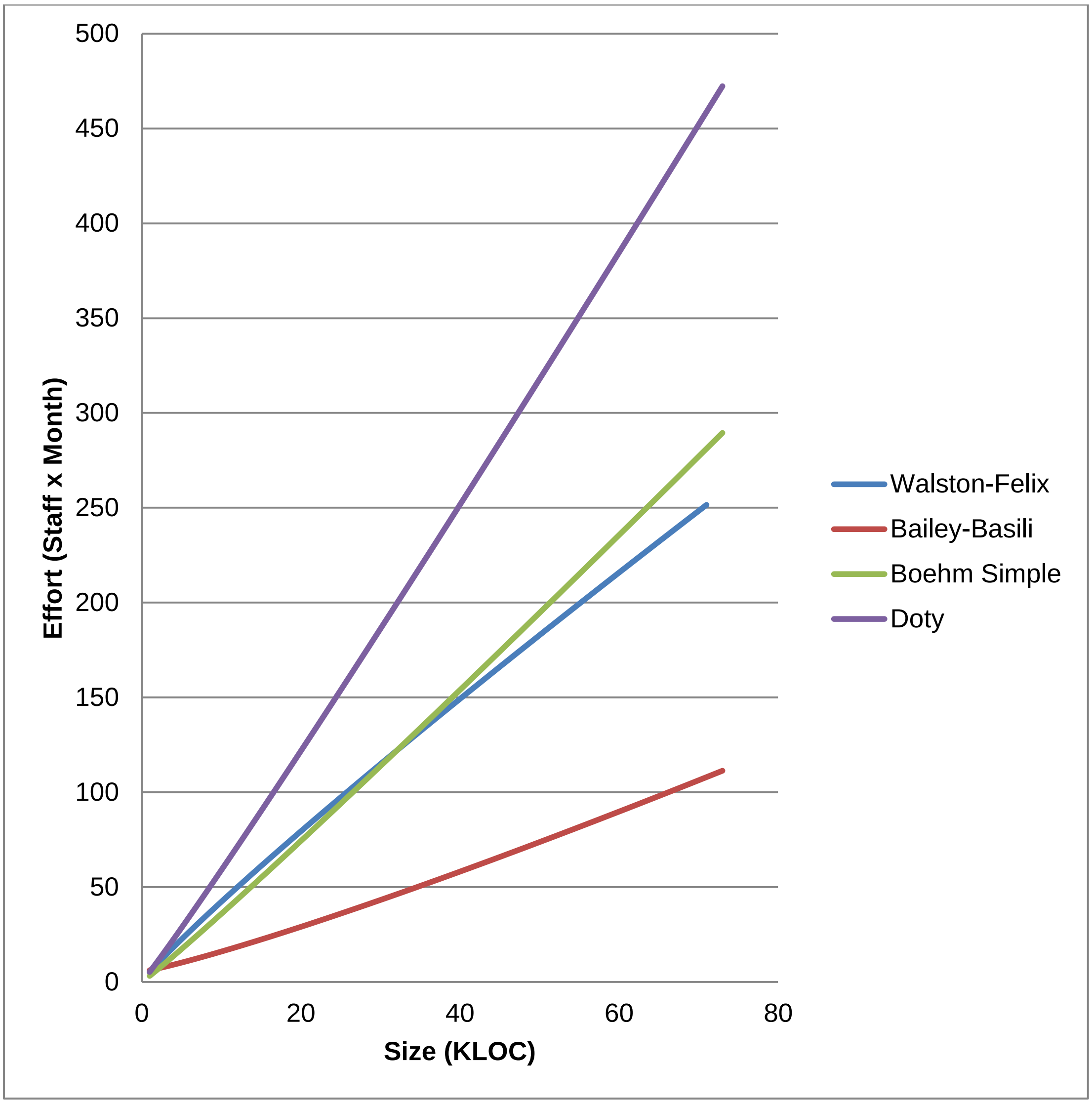

Figure 6 Comparison of Algorithmic KLOC Models

(Source: Laird & Brennan, 2006, p. 104)

Similarly, there are different FP equations calibrated to project data sets. For example: (McGibbon, 1997)

Table 12 Different FP Equations

Equation Name	Equation
Albrect-Graffney	Effort = 13.39 + 0.0545 x $(FP)^1$ (Equation 8)
Kemerer	Effort = 60.62 + 7.728 x 10^{-8} x $(FP)^3$ (Equation 9)
Matson-Berret-Meltichamp	Effort = 585.7 + 15.12 x $(FP)^1$ (Equation 10)

(Source: Laird & Brennan, 2006, p. 103)

3.3.2 COCOMO, COCOMO II and COCOMO 2000

COCOMO was published by Barry Boehm in his book Software Engineering Economics in 1981. COCOCO is an effort, cost and schedule calculation model and it is an algorithmic model. Barry Boehm developed the COCOMO model while he was a director at TRW aerospace division using 63 real life software projects. The projects used as samples range from assembly to PL/I and range in size from 2 KLOC to 100 KLOC and were developed using the waterfall model.

COCOMO is composed of three levels with respect to the detail of estimation:

1. Basic COCOMO
2. Intermediate COCOMO
3. Detailed COCOMO

Basic COCOMO is used for quick estimations where cost drivers are not known. However, since the cost drivers are not included in the estimation process, the error rate of the Basic COCOMO is higher than Intermediate or Detailed COCOMO. Intermediate COCOMO takes cost drivers into account and Detailed COCOMO distributes estimation results over project phases.

3.3.2.1 Basic COCOMO

Basic COCOMO classifies software project in to 3 categories:

1. Organic Project: Organic project is a project type where project staffs are experienced and software requirements are stable.
2. Semidetached Project: Semidetached projects are the kind of projects where project personnel are made up of mixed type of personnel according to their experiences and software requirements are more volatile.
3. Embedded Project: Embedded project is the type of project where tight constraints (hardware, software, performance, operation, etc.) exist.

Basic COCOMO, computes *Effort* and *Development Time* as the function of software size in LOC.

$$Effort = A_{Basic} \times (LOC)^{B_{Basic}} \text{ (Staff x Months)} \quad \text{(Equation 11)}$$

$$Development\ Time = C_{Basic} \times (LOC)^{D_{Basic}} \text{ (Months)} \quad \text{(Equation 12)}$$

$$Staff\ Required = {}^{Effort}/_{Development\ Time} \text{ (Staffs)} \quad \text{(Equation 13)}$$

The parameters A_{Basic}, B_{Basic}, C_{Basic}, D_{Basic} are given in the Table 13 for 3 different project categories. (Boehm, 1981, pp. 57-71)

Table 13 Coefficients of Basic COCOMO Equations

	A_{Basic}	B_{Basic}	C_{Basic}	D_{Basic}
Organic	2,4	1,05	2,5	0,38
Semidetached	3	1,12	2,5	0,35
Embedded	3,6	1,2	2,5	0,32

(Source: Boehm, 1981, p. 75)

3.3.2.2 Intermediate COCOMO

Aside from Basic COCOMO, Intermediate COCOMO computes effort as a function of LOC, and cost drivers. Cost drivers are given in Table 14. Each cost driver is assessed subjectively in order to determine the Effort Adjustment Factor (EAF). The EAF is the product of selected ratings of the cost drivers.

Table 14 Cost Drivers of COCOMO

Cost Drivers	Ratings					
	Very Low	Low	Nominal	High	Very High	Extra High
Product attributes						
Required software reliability	0,75	0,9	1	1,2	1,4	
Size of application database		0,9	1	1,1	1,16	
Complexity of the product	0,7	0,9	1	1,2	1,3	1,65
Hardware attributes						
Run-time performance constraints			1	1,1	1,3	1,66
Memory constraints			1	1,1	1,21	1,56
Volatility of the virtual machine environment		0,9	1	1,2	1,3	
Required turnabout time		0,9	1	1,1	1,15	
Personnel attributes						
Analyst capability	1,46	1,2	1	0,9	0,71	
Applications experience	1,29	1,1	1	0,9	0,82	

(Continued)

Table 14 Cost Drivers of COCOMO *(Continued)*

Cost Drivers	Ratings					
	Very Low	Low	Nominal	High	Very High	Extra High
Software engineer capability	1,42	1,2	1	0,9	0,7	
Virtual machine experience	1,21	1,1	1	0,9		
Programming language experience	1,14	1,1	1	1		
Project attributes						
Application of software engineering methods	1,24	1,1	1	0,9	0,82	
Use of software tools	1,24	1,1	1	0,9	0,83	
Required development schedule	1,23	1,1	1	1	1,1	

(Source: Boehm, 1981, pp. 118-119)

Effort and development time functions of the Intermediate COCOMO are given below:

Effort = $A_{Int.}$ x $(KLOC)^{BInt.}$ x EAF (Staff x Months) (Equation 14)

Development Time = $C_{Int.}$ x $(KLOC)^{DInt.}$ (Months) (Equation 15)

Staff required for the project is calculated as in the Basic COCOMO model. AInt., BInt, CInt. and DInt. are the coefficients of the project types. (Boehm, 1981, pp. 114-141)

Table 15 Coefficients of Intermediate COCOMO Equations

	$A_{Int.}$	$B_{Int.}$	$C_{Int.}$	$D_{Int.}$
Organic	3,2	1,05	2,5	0,4
Semidetached	3,0	1,12	2,5	0,35
Embedded	2,8	1,2	2,5	0,32

(Source: Boehm, 1981, p. 117)

3.3.2.3 Detailed COCOMO

Detailed COCOMO is built on Intermediate COCOMO by introducing additional details of project phases (design, development, test, etc.) and three levels of product hierarchy. The three levels of product hierarchy are System, Subsystem and Module. (Güzel, 2009, p. 36)

3.3.2.4 COCOMO II and COCOMO 2000

COCOMO II and COCOMO 2000 are revised and extended estimation models built on original COCOMO. It uses software size and a set of new factors to calculate effort. The major factors resulting in the drive to update COCOMO were new software processes (spiral, incremental, evolutionary, etc.), new sizing phenomena (function points, use case points), new reuse phenomena (COTS, reusable components) and the need to make decisions based on incomplete information. (Güzel, 2009, pp. 40-41)

3.3.3 SLIM

SLIM is the commercial tool owned by QSM Inc. based upon the Putnam Model. While he was working in US Army projects, later at General Electric, Lawrence H. Putnam noticed a resemblance between the staffing level of the software project and the Rayleigh distribution. The Rayleigh probability density function is given below:

$$P(x) = \frac{x}{\sigma^2} e^{\frac{-x^2}{2\sigma^2}}$$ (Equation 16)

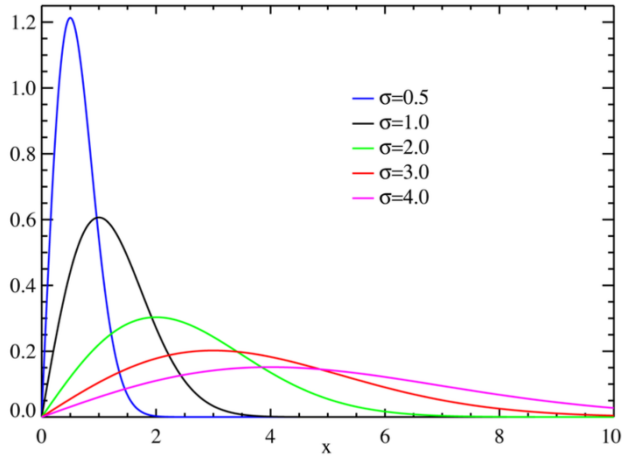

Figure 7 Rayleigh Probability Density Function

(Source: Wikipedia)

Through his observations Putnam derived the following equation:

$$\frac{B^{\frac{1}{3}} \times Size}{Productivity} = Effort^{\frac{1}{3}} \times Time^{\frac{4}{3}}$$ (Equation 17)

where,

B is a scaling factor function of project size, *Size* is project size in LOC, SLOC, FP, etc. appropriate size metrics used by the organization, *Productivity* is the ability of the organization to produce a software at given defect rate, *Effort* is the total effort of the project in staff x years and *Time* is the total schedule time of the software project in the unit of years. (Equation **17** can be rewritten as:

$$Effort = \left[\frac{Size}{Productivity \times Time^{4/3}} \right]^3 \times B$$ (Equation 18)

and also can be rewritten as:

$$\text{Productivity} = \frac{\text{Size}}{\left[\frac{\text{Effort}}{B}\right]^{1/3} \times \text{Time}^{4/3}} \qquad \text{(Equation 19)}$$

Putnam's equations and assumptions have been criticized by a number of researchers. Kitchenham (Kitchenham B. A., Empirical Studies of Assumptions That Underlie Software Cost-Estimation Models, 1992) and Jeffery (Jeffery, 1987) have challenged (Equation 18 that reduction in time increases the effort and vice versa. Moreover Kitchenham and Taylor (Kitchenham & Taylor, Software Cost Models, 1984) found that effort estimation in Putnam's equations is very sensitive to incorrect estimations of both size and productivity.

Parr (Parr, 1980, pp. 291 - 296) criticized the underlying assumptions of Putnam Model. In particular, he criticized the ignorance of a certain level of staffing in the beginning of the project. Parr claims staffing levels which may influence the rest of the project never start from zero, rather they start from certain level of staffing. Parr suggests the sech2 curve which has non-zero intersect with y axis, instead of Rayleigh probability density function.

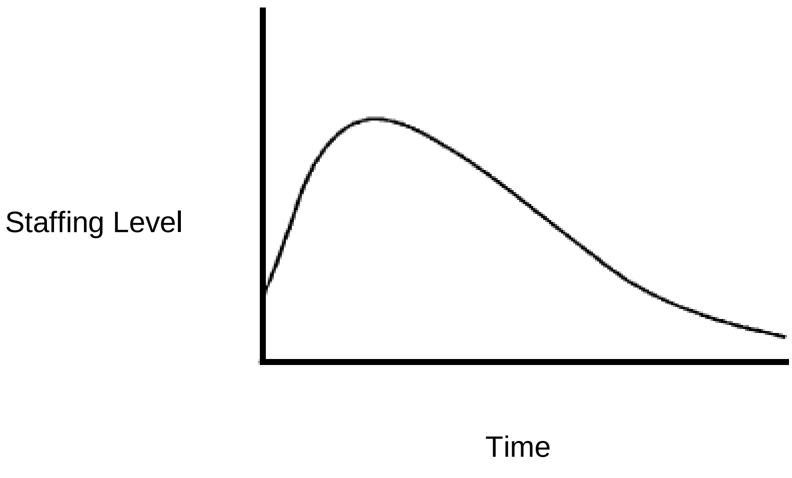

Figure 8 Sech^2 Curve

(Source: Wikipedia)

3.3.4 SEER-SAM

SEER-SEM is commercial software project management tool. This tool includes an algorithmic software cost estimation model which is proprietary and not available to the public as it is owned by Galorath Associates, Inc. The initial version of the application was developed by Galorath in 1988. The project ended with the first version of SEER-SEM software composed of 22.000 LOC. SEER-SEM is designed to use in planning, monitoring and resource estimation of any software development or maintenance projects.

The model used in SEER-SEM is based upon the model developed by Dr. Randall Jensen. Although the mathematical equations used by SEER-SEM are not available to the public, Dr. Jensen's basic equations have been published in order to be reviewed. The basic equation which was named the Software Equation by Dr. Jensen is given below:

$$S_e = C_{te} \times (K \times t_e)^{0,5} \qquad \text{(Equation 20)}$$

where,

S is the *effective lines of code,* C_t is the *effective developer technology constant,* and K is *the total life cycle cost* in man x years and t_d is the *total development time* in years. (Equation 20 relates the system size and the technology that are being applied by the developers. The technology factor is used to calibrate the model to a specific environment. This factor considers two aspects of the production technology which are technical and environment. The technical aspect considers an organization's technical capabilities and maturity, experience of the staff, development practices, tools, etc. The environmental aspect of the technical factor considers environmental conditions in which the software will be deployed, CPU characteristics, network constraints, etc.

SEER-SEM accepts almost 30 input parameters. Moreover, it includes development modes like object oriented programming, COTS, reuse, spiral, incremental and waterfall. It covers 3rd and 4th generation programming languages like C++, FORTRAN, COBOL, Ada, etc. It allows for supplying the capabilities of the staff those would be assigned to the software project. In Figure 9 SEER-SEM software is represented as a black box and possible outputs and inputs of the tool are shown. (Basha & P., 2010, pp. 69-70)

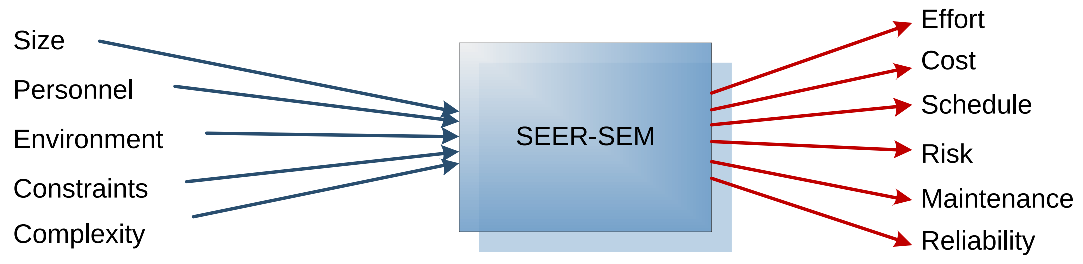

Figure 9 Input and Output Paramters of SEER-SEM

3.4 Proxy Methods

Proxy-Based estimation is an estimation of proxy (for example test cases, screens, etc.) that is correlated with what is really wanted to be estimated. The purpose of Proxy-Based estimation is to overcome challenges of estimating the quantity which is difficult to identify at the beginning of the project by replacing it with something correlated and easier to count. For instance, the number of unique customers can be proxy for estimation LOC.

Function Point, Use Case Point and Story Point are well known Proxy-Based estimation methods

3.4.1 Function Points

Function Point Analysis (FPA) was developed by Allan J. Albrecht in 1979 and gained widespread acceptance in the industry. Nonprofit organizations like IFPUG were established to encourage the use of Function Point Analysis to effectively manage application development and management activities.

The size and cost of the task to develop a computerized system is related to 3 main factors which are *information processing size*, *technical complexity factor* and *environmental factors*. Information processing size is the measure of the information size processed and produced by the system to be developed. The Technical complexity factor is the factor which results when technical difficulties arise from developing or implementing the system. Environmental factors are the factors which arise from the project environment such as the technical skills and motivation of the staff, tools and methods used, development methodologies applied, etc. Albrecht FPA determines the size of the system based on the first two factors (Symons, 1988, pp. 2-3).

Determination of the information processing size starts with identification of the software components as seen by the end users. Then these software components are classified into 5 categories namely *The External Inputs*, *Outputs*, *Inquiries*, *The External Interfaces to Other Systems*, and *the Logical Internal Files*. After this step, each component is classified into 3 categories which are *Simple, Average and Complex*. The sum for all components is defined as the *Unjustified Function Points* (UFP's). Table 16 shows the determination of information processing size.

Table 16 Unjustified Function Point Counting

Description	**Level of Information Processing Function**			**Total**
	Simple	**Average**	**Complex**	
External Input	____x3=____	____x4=____	____x6=____	
External Output	____x4=____	____x5=____	____x7=____	
Logical Internal File	____x7=____	____x10=____	____x15=____	
External Interface File	____x5=____	____ x7=____	____x10=____	
External Inquiry	____x3=____	____ x4=____	____x6=____	
			Total Unjustified Function Points	

(Source: Symons, 1988, p. 3)

The second factor, Technical Complexity Factor, is determined by the Total Degree of Influence composed of 14 different application characteristics. Degree of Influence scale is given below:

i. 0: Not Present, or No Influence
ii. 1: Insignificant Influence
iii. 2: Moderate Influence
iv. 3: Operational Ease, Average Influence
v. 4: Significant Influence
vi. 5: Strong Influence, Throughout

Table 17 Calculation of Total Degree of Influence

ID	**Characteristic**	**DI**	**ID**		**DI**
C1	Data Communications	___	C8	On-line Update	___
C2	Distributed Functions	___	C9	Complex Processing	___
C3	Performance	___	C10	Reusability	___
C4	Heavily Used Configuration	___	C11	Installation Ease	___
C5	Transaction Rate	___	C12	Operational Ease	___
C6	On-Line Data Entry	___	C13	Multiple Sites	___
C7	End User Efficiency	___	C14	Facilitate Change	___
				Total Degree of Influence	

(Source: Symons, 1988, p. 3)

Calculation of the TDI Technical Complexity Factor (TCF) is computed using (Equation **21)** below:

$$TCF = 0.65 + 0.01 \times TDI \qquad \text{(Equation 21)}$$

Finally, size of the system in Function Point is computed by:

$$FS's = UFP's \times TCF \qquad \text{(Equation 22)}$$

The size of the system in Function Points is a dimensionless number in arbitrary scale. Albrecht states the reasons for proposing the Function Point as a measure of the software size as:

1. Function Point Analysis isolates the size of the system from environmental factors, so the study of factors that affect productivity becomes easier.
2. Measure of the system is based on the view of the user and this measurement is technology independent.
3. Measure of the system can be determined in the early stages of the software lifecycle, enabling the Function Points to be used in the estimation process.
4. Function Point Analysis can be understood and used by nontechnical personnel.
5. Function Point Analysis has low measurement overhead.

The effort required to develop a system can be calculated by simply multiplying FP's by average effort to develop a single FP obtained from the software industry. (Albrecht, Allan J.;, 1979, pp. 82-93)

$$\text{Effort} = \text{FP's} \times \text{Effort to Develop Single FP} \qquad \text{(Equation 23)}$$

3.4.2 Use Case Points

The Use Case Points method was developed by Gustav Karner in 1993. It is based on the Function Points method. The purpose of the Use Case Points method is to provide a simple estimation method for software projects where object oriented programming languages are used.

The Use Case Points method requires counting transaction numbers in each use case. A transaction can be defined as an event which occurs entirely of not at all step use of the Use Case Points method are described below:

1. Actors in the use case analysis are classified in to three categories: simple, average and complex. Simple actors can be defined as actors interacting with another system through API, an average actor can be defined as an actor interacting with another system through communication protocols like

TCP/IP, a complex actor can be a person interacting with a system using graphical user interface, web page, etc. The weighting factor for each of the actor categories are as follows:

Table 18 Weighting Factors of Actor Categories

Actor Category	Weighting Factor
Simple	1
Average	2
Complex	3

(Source: Anda, 2002, p. 3)

Unadjusted Actor Weight (UAW) is calculated by multiplying the total number of actors in specific category and summing up these products.

2. Use cases are classified in to three categories as in case of actors: simple, average and complex. Use case which includes transactions of less than 4 falls in to simple category, use case which include numbers of transactions between 4 and 7 fall in to the average category, and use case which includes more than 7 transactions falls in to complex category. The weighting factor for each of the use case categories are as follows:

Table 19 Weighting Factors of Use Case Categories

Use Case Category	Weighting Factor
Simple	5
Average	10
Complex	15

(Source: Anda, 2002, p. 4)

Unadjusted Use Case Weight (UUCW) is calculated by multiplying total number of use cases in a specific category and summing up these products. The UAW is added to the UUCW to get the unadjusted use case points (UUPC).

UUCP=UAW+UUCW (Equation 24)

3. UUCP is adjusted using weight factors Table 20 and Table 21

Table 20 Technical Complexity Factors

Factor	Description	Weight
T1	Distributed system	2
T2	Response or throughput performance objec-tives	2
T3	End-user efficiency	1
T4	Complex internal processing	1
T5	Reusable code	1
T6	Easy to install	0.5
T7	Easy to use	0.5
T8	Portable	2
T9	Easy to change	1
T10	Concurrent	1
T11	Includes Security features	1
T12	Provides access for third parties	1
T13	Special user training facilities are required	1

(Source: Anda, 2002, p. 4)

Table 21 Environmental Factors

Factor	Description	Weight
F1	Familiar with Rational Unified Process	1.5
F2	Application experience	0.5
F3	Object-oriented experience	1
F4	Lead analyst capability	0.5
F5	Motivation	1
F6	Stable requirements	2
F7	Part-time workers	-1
F8	Difficult programming language	-1

(Source: Anda, 2002, p. 4)

Each factor can be assigned a value between 0 and 5; 0 means that the selected factor is irrelevant for the project whereas 5 means this factor is very important. The Technical Complexity Factor (TCF) is calculated as:

$$TCF = 0{,}6 \times \left(0{,}01 \times \sum_{N=1}^{13} T_N \times Weight_N \right) \qquad \text{(Equation 25)}$$

and the Environmental Factor (EF) is calculated as:

$$EF = 1{,}4 \times \left(-0{,}03 \times \sum_{N=1}^{8} F_N \times Weight_N \right) \qquad \text{(Equation 26)}$$

Adjusted Use Case Points (UCP) is calculated as:

$$UCP = UUCP \times TCF \times EF \qquad \text{(Equation 27)}$$

4. Karner proposed planning of 20 staff x hour effort per adjusted use case point. (Karner, 1993)

3.4.3 Story Points

Story Points estimation is an estimation method of Extreme Programming methodology. In this methodology, a project team assigns each story (requirements, features) a number. The numbers can be powers of 2 or Fibonacci numbers.

Table 22 Story Point Scale

Story Point Scale	Specific Points on the Scale
Powers of 2	1, 2, 4, 8, 16
Fibonacci sequence	1, 2, 3, 5, 8, 13

(Source: McConnell, 2006, p. 142)

Example of stories and assigned values are given below:

Table 23 Example of stories and their assigned values

Story	Points
Story 1	4
Story 2	2
Story 3	8
…	…
Total	180

(Source: McConnell, 2006, p. 142)

Story points are not useful at this stage since they are not associated with the number of staff x days, calendar days, LOC, etc. However, the advantage of calculating the total size in points is that the estimation of total size of the project is free of bias.

After the calculation of total project size in Story Points, the project team then plans an iteration covering a specific number of Story Points. After the first iteration is completed the project team has a chance to estimate the total size and schedule of the project much more realistically. The project team can calculate how many story points are completed, how many staff x days are spent, and how much calendar time elapsed. With the help of these calculations, the project team can calculate effort and calendar time required to compete a single Story Point. This is called the *velocity* of the project.

Table 24 Example of Project Velocity Calculation

Data Obtained From Iteration 1
27 story points delivered
12 staff weeks expended
3 calendar weeks expended
Preliminary Calibration
Effort = 27 story points ÷ 12 staff weeks = 2.25 story points/staff week
Schedule = 27 story points ÷ 3 calendar weeks = 9 story points/calendar week

(Source: McConnell, 2006, p. 143)

After the calculation of the project velocity, the project team can estimate the total effort and calendar time required to complete whole project.

Table 25 Initial Projection of the Whole Project

Assumptions (from Preliminary Calibration)
Effort = 2.25 story points/staff week
Schedule = 9 story points/calendar week
Project size = 180 story points
Preliminary Whole-Project Estimate
Effort = 180 story points ÷ 2.25 story points/staff week = 80 staff weeks
Schedule = 180 story points ÷ 9 story points/calendar week = 20 calendar weeks

(Source: McConnell, 2006, p. 143)

However, Story Points estimation is limited to iterative development and extreme programing practices. (McConnell, 2006, pp. 142-143)

CHAPTER 4 SOFTWARE SIZE ESTIMATION PERFORMANCE IN TURKEY

4.1 Methodology

The research in this thesis was done by face to face interviews and by forwarding the electronic form to individuals contacted through social networks. The participants of the survey are employers, managers, directors, program managers, project managers, and team leaders who have project management experience in their professional careers. The participants have sufficient information about sample project characteristics, sample project cost details, team structure and experience, and organizational structure. Moreover, these participants were involved in the cost estimation stage of the sample project. These individuals were selected among highly motivated individuals who were willing to contribute to this thesis research, and they have valuable sight about the Turkish software market and software size estimation practices.

4.2 Survey Design

4.2.1 Survey Questions

The thesis survey questions were grouped into 7 logical parts, labeled A to G. Each part contains specific questions about the following titles: Organizational profile, maturity of the software development lifecycle process, estimation details of the sample project, product constraints, project team experience in product and platform, environmental constraint, and project properties.

4.2.1.1 Part A: Organizational Profile

The questions in Part A focused on how deep rooted the firm is, size (personnel number) of the organization, the company's organizational and shareholder structure, and the sectors that the company is serving (the sectors in the question answers are compliant to TUİK sector grouping). The main purpose of the questions in Part A was to determine any correlation between organizational structure and the estimation performance.

4.2.1.2 Part B: Maturity of Software Development Lifecycle Process

The questions in Part B examined the maturity level of the analysis, design, development, integration and system test process areas of the organization. Each process area was assessed according to one of the five process maturity levels by the survey contributor. Maturity levels start from the level defined as “endeavor of individuals” (similar to hero model in CMMI v1.2), and end with the level defined as “managed, measured and improving process”. The purpose of measuring each of the software lifecycle process phases is to find out the relation between estimation performance and each process phase maturity. Omitting important activities in the process phases such as requirement specifications analysis, and unit, integration and system test activities due to low level of maturity are expected to increase estimation errors. Moreover, process maturities are examined by questioning defined roles in the organization. Undefined roles are assumed to be an indicator of a low level of maturity.

Part B also included a question about the organization software process model, whether it is based on international standards such as CMMI, SPICE, Agile Methodologies, etc. Any compliance with the international standards is assumed to be indication of process maturity.

In addition, software metrics and measurement practices in the organization are questioned in this part. Measurement practices in a specific process phase are assumed to be a “defined process” maturity level in this area.

4.2.1.3 Part C: Estimation Details of the Sample Project

Part C contained two questions. The first question was about the initial effort estimation of sample software project in the unit of staff x days. The second question was about the estimation method(s) applied throughout the estimation process. Here, the survey contributor was warned about the constraints of the screening criteria of the selection of the sample projects. The Projects with the following characteristics were not accepted:

1. Those projects that had not been started or completed in the last 7 years.
2. Those projects that with finished with less than 40 staff x days.
3. The projects that included production of something other than software (i.e. hardware, consultancy, support, service, etc.).
4. The projects that included outsourcing.
5. The projects that were not initially subjected to the estimation process.

4.2.1.4 Part D: Product Constraints

In the Part D, there were questions aimed at software products which were produced with the software project. Product complexity, the level product documentation requirements, the volume of the data stored, percentage of the reused code, strategic importance of the product according to organization, and programming languages used in the production are examined in this part.

Since, there are many formal definitions of metric *software complexity* in the literature, the perception of this metric seems to change from one organization to another, and it is too difficult to measure in practice. Software complexity metric is measured by the time required to adopt a new team member into project. It is assumed that the time required for adaptation of a new member into a project increases with the complexity of the product.

The types of the product documentations were asked in order to determine omitted effort estimation of product documentation activities, especially at the beginning of the project. Since special development requirements of the data intensive systems, size of the data stored (in bytes) is asked. The percentage of the code reused is expected to change coding efforts. Similarly software availability and reliability requirements are supposed to affect project efforts. Stored data size, software availability and reliability requirements were asked in order to determine if there exist correlations between these factors and project effort. The last question, which programming languages are used in the development, was asked to gauge the gearing factor. (See Appendices A for gearing factors of different programming languages)

4.2.1.5 Part E: Project Team Experience in Product and Platform

Part E contained questions aimed at measuring project team properties. The questions were prepared to measure the project team's experience on application, platform, and programming languages used. The unit of experience in a specific field is accepted as the number of similar studies in that field. For instance, the unit of experience in a programming language is the number of projects where that language is used by the project team. The questions were not prepared to measure each individual's experience on a target field; instead it was prepared to measure average experience of project team.

4.2.1.6 Part F: Environmental Constraint

In the Part F, there were questions about the constraints of the environment that the software product produced by the project would be running. The constraints about memory, processor time, communication, and storage capacity are investigated. The answers to the questions in this part are prepared using three main categories: *no limitation*, *somehow limited*, *strict limitations* answers. Environmental constraints are expected to affect development effort of the software product.

4.2.1.7 Part G: Project Properties

In the last Part G, project properties were questioned. In this part, there were questions about start and end date of project, the size of the project according to organization's perception, flexibility of the project schedule, the way that firm won the project, date constraints written on terms of reference or terms of agreement documents.

4.3 Descriptive Statistics

During analysis of the survey results, logarithm of EEA values are used since EEA varies from 0.05 to 15.55. In addition, the direction of the EEA is out of scope of this thesis; absolute value of log (EAA) is used. The average of absolute values of the logarithm of EEA values for small sized, middle sized organizations and for all survey respondents are given in Table 26.

Table 26 Average of Absolute Values of Log (EEA)

Total Number of Sample Projects	Average of Absolute of Log(EEA) (Small Sized Organizations	Average of Absolute of Log(EEA) (Middle Sized Organizations)	Average of Absolute of Log(EEA) (Total)
26	0,71	0,45	0,53

4.3.1 Categorical Question Answers

The categorical question answers, the average of logarithm of EEA for each category and Two-Sample T-Test results of the questions that have two possible answers are given in APPENDIX D. Since the total number of sample projects is 26, questions which have two possible answers and have answer distribution of 22-4 to 26-0 are omitted. .

4.3.2 Cross Tabular Statistics of Survey Answers

Cross tabular statistics for survey answers and firm size are given in APPENDIX E. Columns headers labeled "Middle Size" in the tables in APPENDIX E represent organizations which have a number of employees greater than 50 and smaller than 250, column headers labeled "Small Size" represents organizations which have a number of employees smaller than 50. The cross section cell of a question row and an organization size column contains two numbers. The upper floating point number is the average of the absolute value of the logarithm of the EEA values. The lower integer number is the number of the answers.

4.3.3 Regression Analysis of Survey Results

Categorical predictors are represented by dummy variables as follows:

1. Categorical questions with “Yes” or “No” answers are represented by dummy variables of 0 and 1. a “No” answer is represented by number 0; a “Yes” answer is represented by number 1.
2. Categorical questions with answers more than 2 are represented by numbers 1, 2, 3 … For example, 1 = Disagree, 2 = Neutral, 3 = Agree, etc.

Table 27 Regression Model - 1

Model Details					Model
Predictor	**Coef.**	**SE Coef.**	**T-Value**	**P-Value**	|Log(Estimation Accuracy)| =
Constant	-0,0799	0,2224	-0,360	0,724	- 0,080
No Reference Model	0,5603	0,137	4,090	0,001	+ 0,560 No Reference Model
Documentation of Design	0,3017	0,1056	2,860	0,010	+ 0,302 Documentation of Design
Limitation of CPU Time in Prod. Environment	0,6022	0,1264	4,760	0,000	+ 0,602 Limitation of CPU Time in Prod. Environment
Limitation of Storage Capacity in Prod. Environment	-0,5490	0,1417	-3,880	0,001	- 0,549 Limitation of Storage Capacity in Prod. Environment
Measurement of Time Estimated / Spent per Project Stage	0,5469	0,1477	3,700	0,002	+ 0,547 Measurement of Time Estimated/Spent per Project Stage
Log (Number of Employees)	-0,1017	0,1016	-1,000	0,329	- 0,102 Log (Number of Employees)
S = 0,233189 R^2 = 72,7% Adjusted R^2 = 64,0%					

Table 28 Regression Model - 2

<table>
<tr><th colspan="5">Model Details</th><th>Model</th></tr>
<tr><th>Predictor</th><th>Coef.</th><th>SE Coef.</th><th>T-Value</th><th>P-Value</th><td rowspan="8">|Log(Estimation Accuracy)| =
0,027
+ 0,565 No Reference Model
+ 0,335 Documentation of Design
+ 0,586 Limitation of CPU Time in Prod. Environment
- 0,541 Limitation of Storage Capacity in Prod. Environment
+ 0,458 Measurement of Time Estimated/Spent per Project Stage
- 0,0974 Log (Estimated Effort)</td></tr>
<tr><td>Constant</td><td>0,0272</td><td>0,2998</td><td>0,090</td><td>0,929</td></tr>
<tr><td>No Reference Model</td><td>0,5649</td><td>0,1345</td><td>4,200</td><td>0,000</td></tr>
<tr><td>Documentation of Design</td><td>0,3354</td><td>0,1100</td><td>3,050</td><td>0,007</td></tr>
<tr><td>Limitation of CPU Time in Prod. Environment</td><td>0,5861</td><td>0,1271</td><td>4,610</td><td>0,000</td></tr>
<tr><td>Limitation of Storage Capacity in Prod. Environment</td><td>-0,5412</td><td>0,1426</td><td>-3,790</td><td>0,001</td></tr>
<tr><td>Measurement of Time Estimated / Spent per Project Stage</td><td>0,4580</td><td>0,1620</td><td>2,830</td><td>0,011</td></tr>
<tr><td>Log (Estimated Effort)</td><td>-0,0974</td><td>0,0945</td><td>-1,030</td><td>0,316</td></tr>
<tr><td colspan="5">S = 0,232840 R^2 = 72,7% Adjusted R^2 = 64,1%</td><td></td></tr>
</table>

Table 29 Regression Model - 3

Model Details

Predictor	Coef.	SE Coef.	T-Value	P-Value
Constant	-0,1357	0,2036	-0,67	0,513
No Reference Model	0,6279	0,1265	4,96	0,000
Documentation of Design	0,3799	0,1438	2,64	0,016
Limitation of CPU Time in Prod. Environment	0,5697	0,1337	4,26	0,000
Limitation of Storage Capacity in Prod. Environment	-0,5306	0,1495	-3,55	0,002
Measurement of Time Estimated / Spent per Project Stage	0,5357	0,1483	3,61	0,002
Log (Project Team Size)	-0,1909	0,2397	-0,80	0,436

S = 0,235361 R^2 = 72,1%   Adjusted R^2 = 63,4%

Model

|Log(Estimation Accuracy)| =
- 0,136
+ 0,628 No Reference Model
+ 0,380 Documentation of Design
+ 0,570 Limitation of CPU Time in Prod. Environment
- 0,531 Limitation of Storage Capacity in Prod. Environment
+ 0,536 Measurement of Time Estimated/Spent per Project Stage
- 0,191 Log (Project Team Size)

Table 30 Regression Model - 4

Model Details					Model
Predictor	**Coef.**	**SE Coef.**	**T-Value**	**P-Value**	|Log(Estimation Accuracy)| =
Constant	-0,15020	0,25700	-0,58	0,566	- 0,150
No Reference Model	0,62380	0,12820	4,86	0,000	+ 0,624 No Reference Model
Documentation of Design	0,31830	0,11380	2,80	0,011	+ 0,318 Documentation of Design
Limitation of CPU Time in Prod. Environment	0,61440	0,13260	4,63	0,000	+ 0,614 Limitation of CPU Time in Prod. Environment
Limitation of Storage Capacity in Prod. Environment	-0,57630	0,14440	-3,99	0,001	- 0,576 Limitation of Storage Capacity in Prod. Environment
Measurement of Time Estimated / Spent per Project Stage	0,51610	0,15300	3,37	0,003	+ 0,516 Measurement of Time Estimated/Spent per Project Stage
Log (Actual Effort)	-0,03304	0,07812	-0,42	0,677	- 0,0330 Log (Actual Effort)

S = 0,238139 R^2 = 71,5%   Adjusted R^2 = 62,5%

Table 31 Regression Model - 5

Model Details					Model
Predictor	**Coef.**	**SE Coef.**	**T-Value**	**P-Value**	| Log(Estimation Accuracy) | =
Constant	-0,2110	0,1974	-1,07	0,299	- 0,211
No Reference Model	0,5999	0,1522	3,94	0,001	+ 0,600 No Reference Model
Documentation of Design	0,3032	0,1082	2,80	0,011	+ 0,303 Documentation of Design
Limitation of CPU Time in Prod. Environment	0,5942	0,1341	4,43	0,000	+ 0,594 Limitation of CPU Time in Prod. Environment
Limitation of Storage Capacity in Prod. Environment	-0,5620	0,1473	-3,82	0,001	- 0,562 Limitation of Storage Capacity in Prod. Environment
Measurement of Time Estimated / Spent per Project Stage	0,5276	0,1506	3,50	0,002	+ 0,528 Measurement of Time Estimated/Spent per Project Stage
Definition of Quality Management	-0,0259	0,1235	-0,21	0,836	- 0,026 Definition of Quality Management
S = 0,238980 R^2 = 71,3% Adjusted R^2 = 62,2%					

Table 32 Regression Model – 6

Model Details					Model
Predictor	**Coef.**	**SE Coef.**	**T-Value**	**P-Value**	| Log(Estimation Accuracy) | =
Constant	-0,1106	0,2064	-0,54	0,598	- 0,111
No Reference Model	0,5726	0,1335	4,29	0,000	+ 0,573 No Reference Model
Documentation of Design	0,2893	0,1068	2,71	0,014	+ 0,289 Documentation of Design
Limitation of CPU Time in Prod. Environment	0,6089	0,1269	4,80	0,000	+ 0,609 Limitation of CPU Time in Prod. Environment
Limitation of Storage Capacity in Prod. Environment	-0,5641	0,1407	-4,01	0,001	- 0,564 Limitation of Storage Capacity in Prod. Environment
Measurement of Time Estimated / Spent per Project Stage	0,4837	0,1546	3,13	0,006	+ 0,484 Measurement of Time Estimated/Spent per Project Stage
Percentage of Total Effort of Project Team	-0,03804	0,0399	-0,95	0,352	- 0,0380 Percentage of Total Effort of Project Team
S = 0,233731 R^2 = 72,5% Adjusted R^2 = 63,9%					

4.3.4 Scatter Plot of Continuous Data

Scatter plot of continuous survey data and the linear regression lines are given below.

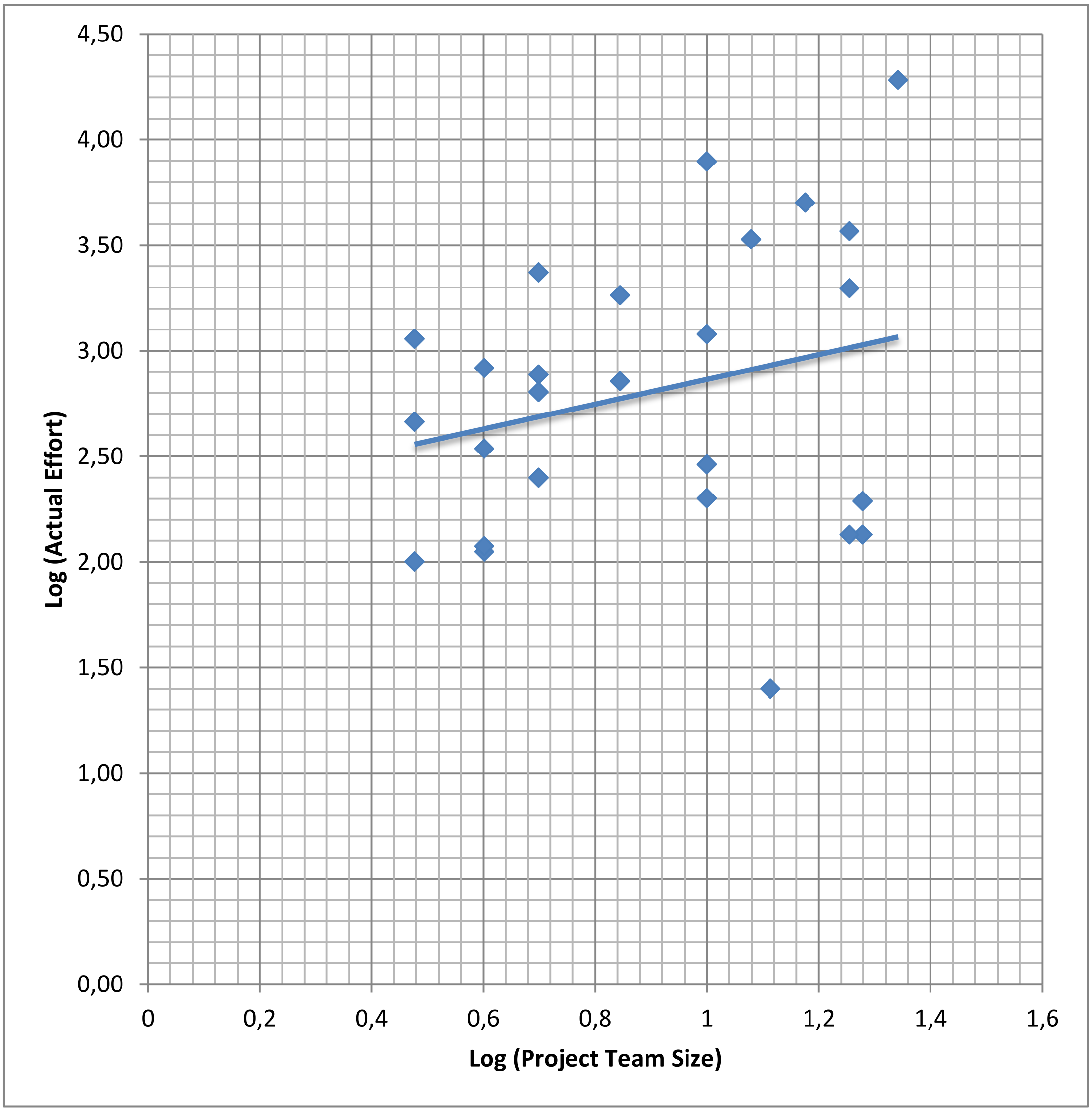

Figure 10 Scatter Plot of Logarithm of Actual Effort vs Logarithm of Project Team Size

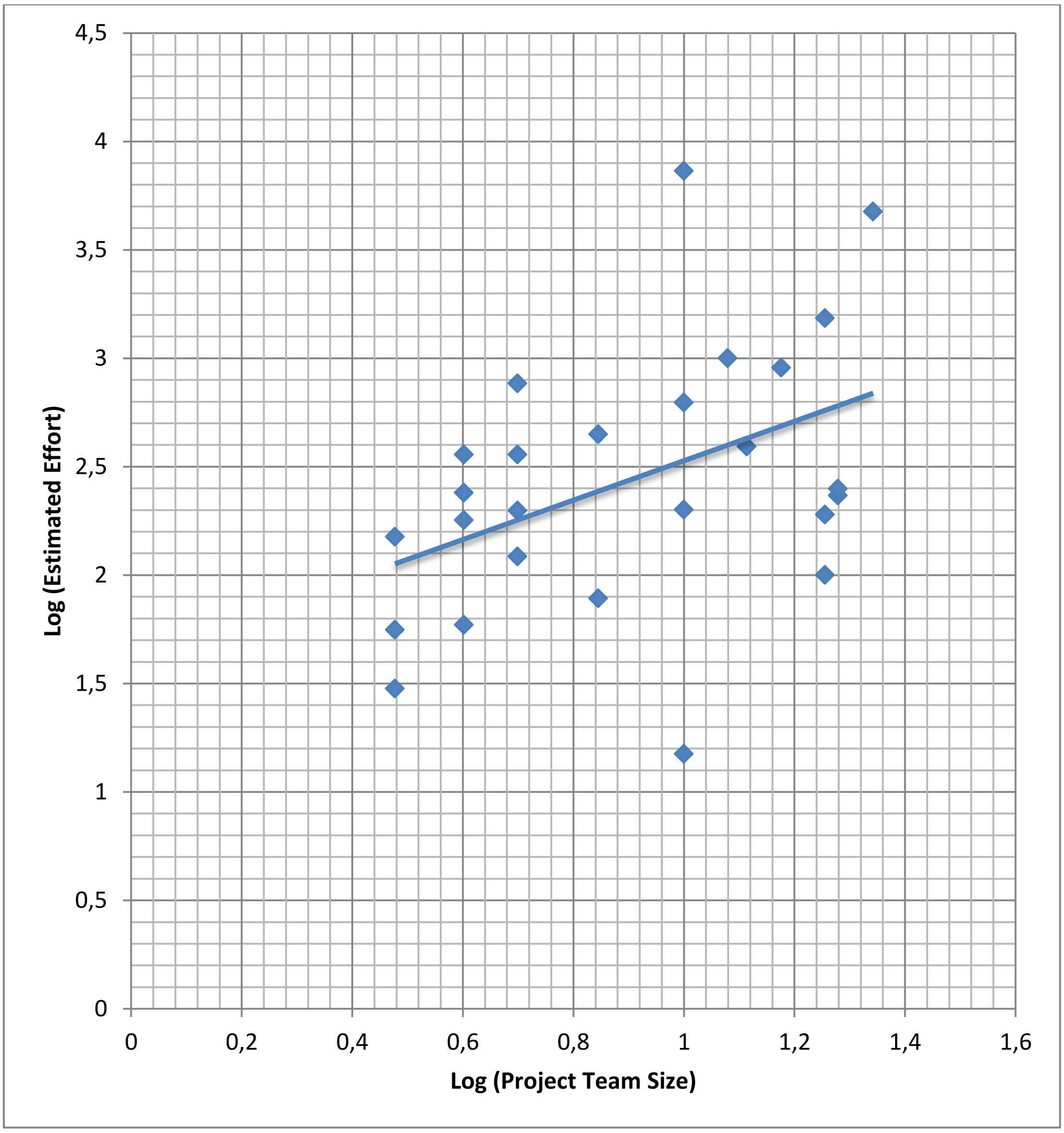

Figure 11 Scatter Plot of Logarithm of Estimated Effort vs Project Team Size

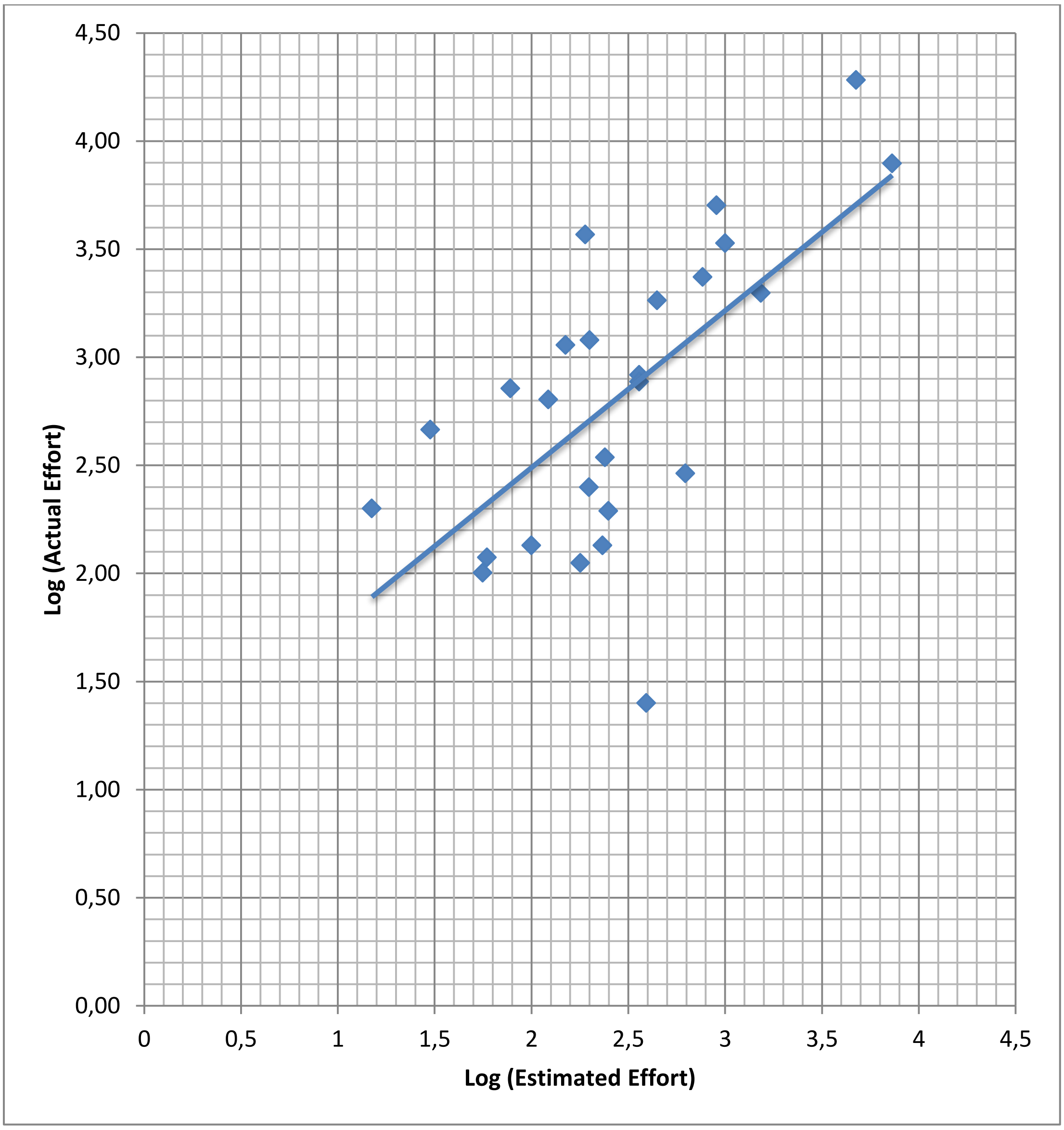

Figure 12 Scatter Plot of Logarithm of Estimated Effort vs Logarithm of Actual Effort

4.4 Software Size Estimation Performance

26 survey results were obtained through face to face interviews and electronic form submissions. The actual efforts spent in the sample projects were obtained by two different methodologies. The actual efforts were obtained from the contributors whose organizations kept records of effort spent by the employees. On the other hand, actual effort spent is calculated by the (Equation 28 in the organizations where efforts spent by the employees are not recorded nor reported.

Calculated Actual Effort = Project Time x Number of Staff Assigned (Equation 28)

In the formulation process of (Equation 28 it was assumed that, the entire project staffs spent their efforts on the project from the beginning to the end Calculated actual effort is always expected to be greater than real life value. Since project team changes over time and project team members could never spend 100 % of their efforts on a single project.

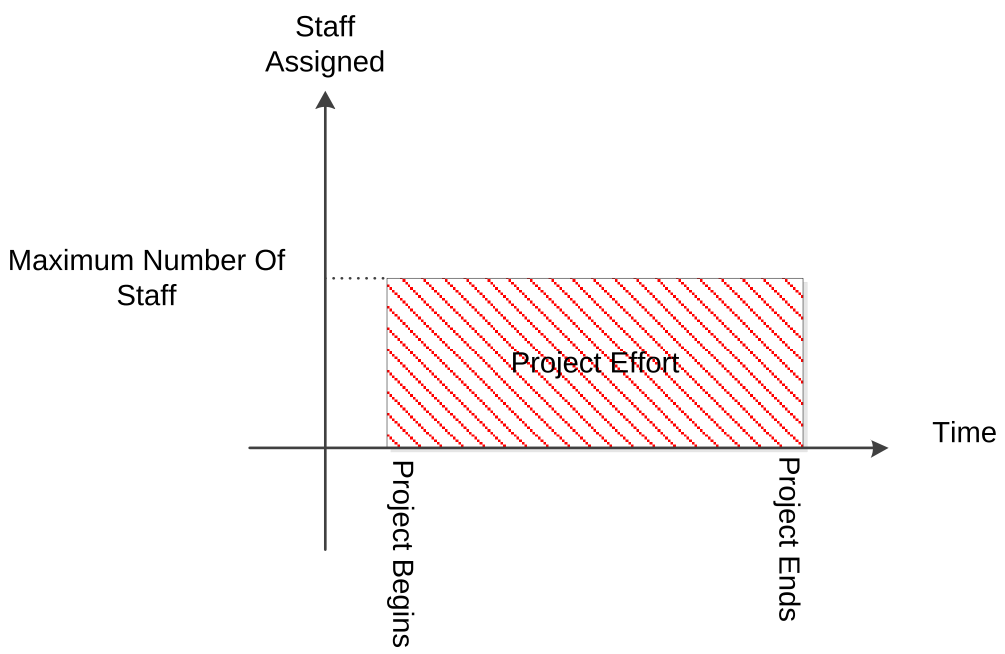

Figure 13 The Method of Effort Estimation of Sample Projects

EEA is calculated by (Equation 29) given below:

$$EEA = \frac{\text{Estimated Project Effort}}{\text{Actual Project Effort}} \qquad \text{(Equation 29)}$$

For better effort estimations, EEA should converge to 1 or alternatively the absolute value of Log_{10} (EEA) should converge to 0. Log function is used as the short hand notation of the Log_{10} function in the entire thesis.

4.4.1 Analysis of the Survey Results

Some important results can be derived from analysis of tables in APPENDIX D and APPENDIX E One of the most remarkable results is that middle sized organizations have better average EEA values than small sized organizations (Table D-1). This result is not surprising because middle sized organizations have more resources than small sized organizations to spend on estimation practices. Since the success of software estimation depends mainly on the effort and time spent to make it more accurate, middle sized organizations are more successful than small sized organizations. In addition, estimation accuracy grows with the accumulation of past project data. Small sized organizations are often new or lack the resources to keep records of past project data. Thus, small sized organizations usually have little past project data. Another remarkable result is that, international organizations have more accurate EEA than local software firms (Table D-2). This result is also an expected result, because estimation practices are very new to Turkish Software sector. However, foreign software researchers and practitioners have been addressing the problems of effort estimation in software development projects since at least the 1960s. The advancement of international organizations on the subject of software estimation is an inevitable consequence

The survey results indicate that average EEA values change depending on which sector the organization provides services or product. For instance, the organizations who provide services and products for food, drink and tobacco production, construction, transportation, storage and communications, finance, real estate, rental and business activities sectors have relatively better average EEA values than the organizations who provide services and products for textile and textile production, main metal and fabrication metal product production, electrical and optical hardware production, wholesale and retail trade, maintenance of motor vehicle, motorcycle, personal goods and furniture, institutions that operate in the fields of public administration and defense, obligatory social insurance sectors. Nevertheless it is not possible to assert that sectors directly affect the performance of the estimation practices since there is not enough survey data prove that hypothesis (Table D-7).

Definitions or System Test and Integration, System and Software Maintenance processes seem to have trivial effects on EEA (Table D-6, Table D-8). Yet, definitions of Quality and Risk Management processes result in considerable enhancement in EEA (Table D-9, Table D-10). The definition of Quality Management in the organizational level enforces not only the quality of the product but also compliance with the institutional processes. Defined and managed processes uncover cost items in the software project. Similarly, the Risk Management process requires detailed analysis of unforeseeable incidents and changes that may occur in the duration of project lifetime. Detailed investigation and analysis of uncertain issues also uncovers cost items in the software project. In summary, Risk Management prevents bad surprises. In addition to these, it can be said that the Definition of Subcontractor Management process enhances EEA (Table D-10); however it is not as influential as Quality Management and Risk Management processes.

The answers of the survey questions aimed at determining the maturity level of project management, requirement management, software design, software development processes cumulate on level 2 (managed level) and level 3 (managed and defined in enterprise level). There is not enough evidence to conclude that there is a correlation between EEA and the maturity levels of these processes. Nonetheless, organizations in the maturity level 2 have better average EEA values than organizations in the maturity level 3 with reference to project management, requirement management, software design and development processes (Table D-12, Table D-13, Table D-14, and Table D-15). Indeed, worse average EEA values in the maturity level 3 arise from EEA values of small sized organizations. A detailed examination of average EEA values of middle sized organizations shows that there are trivial differences between level 2 and level 3 maturity levels (Table E-11, Table E-12, Table E-13, and Table E-14).

If it is taken into account that corresponding average EEA values of each answer of the survey questions aimed at assessing the maturity level of software testing and software maintenance processes, it can be seen that average EEA values evoke improving EEA (Table D-16, Table D17) Although, there is not enough survey data to prove that EEA improves with the maturity level of software testing and software maintenance process, it can be claimed that small sized organizations which have maturity level 1 (process sustained by personal endeavor) have the worst average EEA values (Table E-15, Table E-16).

Organizations which base their software development lifecycle process on CMMI or SW/CMM have better EEA than organizations which do not base their development process on CMMI or SW/CMM (Table D-18). Almost all of the middle sized organizations base their software development lifecycle process on CMMI or SW/CMMI (17 out of 18 organizations). In contrast, almost none of the small sized organizations used CMMI or SW/CMMI (7 out of 8 organizations) in their software development lifecycle process (Table E-17). CMMI or SW/CMM is widely accepted, at the same time relatively expensive reference model since it required certain activities to be performed, and certain traces to be left behind. The cost of the CMMI or SW/CMM reference model cannot be covered by small sized organizations. But organizations that have enough resources to use CMMI or SW/CMM reference model could determine cost items more easily than other organizations. In addition, survey results points out that, organizations which do not base their software development lifecycle process on any reference model have worse average EEA than other organizations (Table D-20), and these five organizations are all small sized organizations (Table E-19). Survey results show that refusing to use widely accepted reference models can be more expensive than implementing them. Organizations that are not using reference model have average EEA value of 0,81 which is worse than both the average of small sized organizations and average of all respondents. Other types of software development lifecycle process models such as Agile Methodologies, SPICE, etc. are not widely accepted models among the respondents included in this survey. Thus, it was not possible to observe the impact of other types of reference models on estimation performance.

Survey results indicate that average EEA values of the organizations which have customer or agreement manager, requirement engineer or analyst, architect, quality manager or engineer, configuration manager roles defined, are better than average EEA values compared with those who do not have them defined in enterprise level (Table D-21, Table D-22, Table D-23, Table D-26, and Table D-27). The definition of the roles can be thought as the evidence for the related processes are carried out. But, having the graphical user interface designer, test engineer roles defined or not, seems to be inconclusive on average EEA values of organizations (Table D-24, Table D-25).

Answers to the questions which attempted to reveal the relation between organizational structure and EEA show that more flexible structures like temporary structures or matrix structures have better average EEA than more static structures like hierarchic or project management oriented structures for all organizations (Table D-28). However, detailed analysis of the results indicates that structuring with respect to software project needs enhances the EEA of middle sized companies. In addition, hierarchical structures are the worst structures for middle sized organizations from the estimation accuracy point of view. Although there are only two respondent organizations (one is small and the other is middle sized organization) that have temporary organizational structure, they have the best average EEA values. Yet, in this answer small sized organizations are neglected, and it can be said that hierarchical structures are performing well with respect to EEA (Table E-26). This result is an expected result since a hierarchical structure is a reasonable structure for small sized organizations having less than 50 employees. Furthermore, a limited number of employees prevents designing temporary project teams with respect to project needs.

Survey results show that software measurement practices in an organization have neither negative nor positive effects or no effect on EEA. Results indicate that middle sized organizations which are measuring the number of requirements have worse average EEA values than middle sized organizations not measuring them.

Middle sized organizations which are not measuring the number of requirements have better than average EEA. Measurement practices of the number of requirement are unrelated to EEA in small sized organizations (Table D-29, Table E-27). Organizations concentrating on the number of requirements might be missing software metrics of design and development stages of the projects. Missed software project stages are shown in Figure 14. Moreover, organizations measuring the number of requirements might be correlating the number of requirements and development effort linearly. However, linear correlation results in oversight of extra effort caused by the diseconomies of the scale.

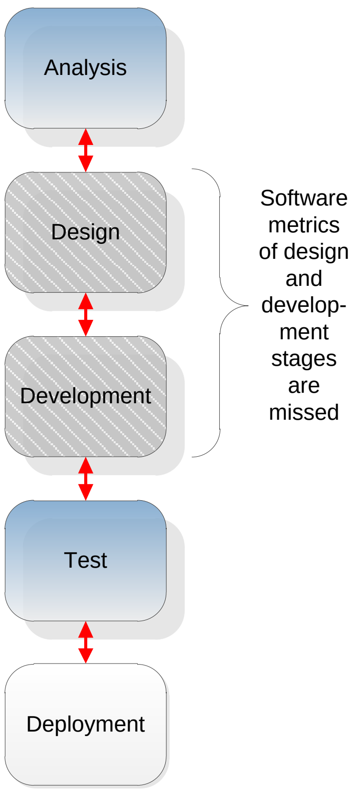

Figure 14 Missed Project Stages

Software defects measurement practice does not seem to have an effect on overall EEA (Table D-30).Nonetheless; it enhances average EEA of smalls sized organizations (Table E-28). Furthermore, measurement practice of software metric of effort by activity worsens for both small and middle sized organizations (Table D-31, Table E-29). Effort by activity software metric can be misleading if coherence of the activities with project phases is not controlled. For instance, if someone in the project team is changing his or her efforts on analysis activities while the project is in the development stage, it demonstrates that business analysis stage is incomplete and some rework is being done. The early impression that the survey results left is that there is a positive effect of software size measurement practices on overall EEA values (Table D-32). However, detailed analysis of the results shows that small sized organizations do not measure software size and measurement of software size does not affect average EEA values of middle sized organizations (Table E-30). Software metric of staff estimated or assigned per project phase is trivial for middle sized organization whereas it negatively affects average EEA values of small sized organizations (Table D-33, Table E-31). Moreover, software metrics of time estimated/realized and cost estimated/realized are trivial for middle sized organizations in accordance with EEA. In contrast, these two metrics enhance average EEA values of small sized organizations (Table D-34, Table D-35, Table E-32, and Table E-33). Time and cost are more important factors for small sized organizations with limited resources than middle sized organizations. In order to complete software projects on time with very limited budgets requires monitoring cost and time spent. Continuous measurement of cost and time while the project is being carried out prevents budget and schedule overruns.

Budget limitations preventing measurements practices from being adopted enhance average EEA values for both small and middle sized organizations (Table D-36, Table E-34). Insufficient support from senior management and reluctance of employees that prevent adopting measurement practices are irrelevant to EEA values of small sized companies. In contrast, these obstacles improve the average EEA of middle sized organizations (Table D-37, Table D-38, Table E-35, and Table E-36). Lastly, inconvenient organizational culture preventing measurements to be adopted worsen the average EEA of small sized companies while it does not affect the average EEA middle sized companies (Table E-38). The habitat, conventions and culture of the small sized organizations are highly influenced by senior managers and employers. Inconvenient culture, sense of rule and qualification of senior managers and employers worsen the EEA of small sized organizations.

Usage of expert judgment estimation method seems to worsen the average EEA of middle sized organizations slightly, however, it enhances dramatically the average EEA of small sized organizations (Table D-41, Table E-39).Compared to the expert judgment method, use of similarity method enhances average EEA of both small and middle sized companies (Table D-42, Table E-40). The practice of estimation by decomposition of tasks methodology results in worse average EEA values in middle sized companies yet it seems to affect the average EEA of small sized companies positively (Table D-43, Table E-41). The decomposition of systems estimation method affects the average EEA of both small sized and middle sized companies (Table D-44, Table E-42). The average EEA of companies using expert judgment and similarity method is better than companies using decomposition of tasks or systems (Table D-41, Table D-42, Table D-43, and Table D-44). Decomposition of tasks or systems sometimes leads to omitting critical activities listed in Table C-1. In specific, decomposition of systems with system breakdown perspective brings about inadequate estimates. However, survey results showed that expert judgment and similarity methods are the most successful estimation methods.

Survey results revealed that documentation practices of specific process areas sometimes improves EEA, sometimes worsens EEA depending on type of the documentation practice and size of the organization. Documentation of contract management worsens the average EEA values of both small and middle sized companies (Table D-46, Table E-44).In addition, documentation of design practices in the sample project improves the average EEA values of small sized organizations while, it affect negatively the average EEA of middle sized companies. It can be said that documentation of design practice worsens EEA values (Table D-47, Table E-45). Documentation of test cases points out that EEA might be improving if the documentation of the test cases realized. However, there are not enough survey results to prove this claim. But, it can be said that documentation of test cases worsens the average EEA values of middle sized companies (Table D-48, Table E-46).Yet, documentation of test results enhances the average EEA for both small and middle sized organization. Furthermore it dramatically enhances the average EEA of small sized organizations (Table D-49, Table E-46).Documentation of user manuals does not affect the average EEA in small sized organization whereas it worsens the average EEA values in middle sized organizations.

Corresponding average EEA values of answers of the questions targeted to determine storage requirements of the product and code reused in the development do not fit the pattern for small sized organizations. However, it can be claimed that, storage requirement and code reuse does not seem to affect the average EEA values of middle sized organizations (Table D-51, Table D-52, Table E-49, and Table E-50).

One of the interesting results of the survey is the relation between the average EEA values and programming languages. Usage of Java programming language enhances the average EEA values for small sized companies whereas usage of C# programming language enhances the average EEA values of middle sized companies. The opposite is also true. Usage of Java programming languages worsens he average EEA values of middle sized companies; in contrast usage of C# programming language worsens the average EEA values of small sized organizations (Table D-56, Table D-57, Table E-54 and Table E-55). The contrary relationship between Java and C# programming languages seems to result from the nature of these two programming languages. Java is an object oriented programming language developed by Sun Microsystems. Software applications developed with Java run on any operating system as long as a Java virtual machine is installed on it. Java became the primary platform of open source supporters in the past 15 years because of Java's OS independence. Development tools of Java such as coding environments can be obtained without any cost. There are a wide variety of free open source Java components available on the Internet that can be used in software projects. The characteristics of Java make it more preferable than other programming languages especially for small sized organizations with very limited resources since it is possible to start a software project on Java platform with very little investment. However, scattered information about Java makes it more difficult to learn and hire a new developer or train an existing one. On the contrary, C# is an object oriented programming language developed by Microsoft. Software applications developed with C# run only on Microsoft Windows OS as long as .Net Framework is installed. C# is not a platform independent programming language. Although there are many C# development tools which are free, it is not easy to develop enterprise applications with these tools. C# development tools require initial investment in development tools. However, OS dependency and centralized SDK tools and the help of C# make it easier to learn and hire a new developer or train an existing one. Hence, C# seems to be preferred programming language for middle sized organizations which can afford development tools, whereas Java seems to be preferred programming language for small sized organizations which cannot afford expensive development tools. Thus, middle sized organizations estimate with lower error ratios if they use C#, small sized organizations estimate with lower error ratios if they use Java. Furthermore, the COBOL programming language is not used in small sized companies and middle sized companies using COBOL and middle sized companies not using this programming language have the same average EEA values (Table D-58, Table E-56). COBOL programs are mainly used in the finance sector and they run on very expensive mainframe systems. This can be the reason why COBOL programming language is not used in small sized organizations. Also, the invariability of the average EEA values for middle sized organizations using the COBOL programming language and those are not using COBOL in their projects can be explained by more predictable nature of COBOL programming language. The COBOL programming language and environment on which COBOL programs run are more predictable than other systems since it has been in use for a longer time than either Java or C#. Usage of other type of programming languages not mentioned in the survey question enhances the average EEA values of middle sized organizations while it worsens the average EEA values of small sized companies (Table D-59, Table E-57).

It is not possible to assert that the percentage of the effort spent by the project team for the sample project has a linear relation with the average EEA of small sized organizations. However, it can be proposed that if the project team spends more than 40 percent of their total effort, the average EEA values of the middle sized organizations are gradually enhanced (Table D-61, Table E-59). Another important statistic can be deduced from Table E-59. 14 of 26 respondents (54 % of total respondents) reported that project teams spent less than 60 % of their effort for the sample project. This statistic shows that on 54% of all projects project team members are assigned to another project or they spent more than 40 % of their effort for maintenance tasks.

The limitations in the production environment seem to have positive effect on the average EEA values for both small and middle sized organizations. Memory limitations, CPU time usage limitations, and communication bandwidth usage limitations enhance the average EEA values for both small and middle sized organizations (Table D-65, Table D-66, Table D-67, Table E-63, Table E-64, and Table E-65). Storage capacity limitations seem to affect the average EEA values for only small sized organizations (Table D-68, Table E-66). Physical limitations lead to detailed analysis and design of software product at the beginning of the project to fulfill production environment limitations. Detailed design and analysis can uncover technical difficulties, obstacles which will emerge in the future and bring about cost or schedule overruns.

CHAPTER 5 CONCLUSION

Survey results showed that the EEA of the sample software projects vary in a wide range. For instance, the logarithm of the EEA values of the sample projects vary in the range of 0,03 to 1,29 which is relatively wide. However, values obtained from survey results can be more optimistic than real life situations. Although overtime is a very common practice, payment of overtime is not a common practice among Turkish software firms. Extra effort spent because of the overtime in order to complete the software projects at planned date could be omitted in the calculations of actual efforts of the sample software projects. For example, in a software project in which extra 2 staff x hours spent by each project team member per day on average means 25% higher project cost than actual project cost in a software firm where official working duration is 8 hours. Actual costs declared in the survey answers are thought to be less than real life data. Therefore, it would not be incorrect to conclude that, estimation performance of the Turkish software sector could be worse than results of this thesis study.

Survey results also showed that, programming languages used in the software project directly affects EEA. Small sized firms using Java programming language and middle sized firms using C# in their projects have better EEA. This result should be explained with special conditions of the software firms and the programming language characteristics. Relatively better EEA in small sized company using Java programming language and middle sized company using C# programming language are thought to be related to development tools, development environments and software libraries. It was observed that, small sized companies are more experienced with Java software development tools and environments, and software libraries which are open source and free. This experience of the small sized firms seems to result in better effort estimation when it comes to estimation of software project effort in which Java programming language is used. However, use of open source but commercial software development tools and libraries increases as the size of the project and the firm increase. At the same time, total effort of the software project increases because of the time spent in the learning of the new Java development tools and libraries. C# programming language is not preferred by the small sized companies because the initial cost of C # programming tools and software libraries are relatively higher than free open source tools and libraries. On the other hand, as the size of the organization increases, the software firms start to invest in Microsoft platforms and tools. Investment in Microsoft tools and platforms increases the experience on C# programming language. It can be concluded that increase in the experience of development with C# programming language yields better estimation results for software projects in which C# programming language is used in middle sized software firms.

Survey results revealed that the widely used estimation method is expert judgment method. Nevertheless, except for Delphi and Wideband Delphi Methods, estimation methods based on expert judgment are open to political pressure and bias. Estimated efforts of the sample project in which expert judgment is used might be subject to pressure from upper management or dominant individuals of project team to reduce the estimation values because of the negotiation of the estimation result. Therefore, estimation performance of sample project in which expert judgment is used could be better than survey results if this assumption is correct.

It was observed that foreign software firms do not have a direct or indirect effect on software size estimation performance of the local firms in Turkey since foreign firms sell only their COTs and they do not develop their software products in Turkey. The foreign software firms do not transfer know-how about software estimation practices to the local software firms. In this respect, globalization has negative affect on local software firms in the competition with the foreign software firms.

Thesis research showed that the best estimation methods among all methods in the question are expert judgment and similarity methods. It is also shown that COCOMO, tool aided estimation methods and proxy methods are not widely known and used estimation methods by survey contributors. Even the Delphi method is rarely applied despite its effectiveness. It is also shown that the software development lifecycle process definitions and process roles definitions are important for success of the estimation practices. On average, middle sized organizations have better EEA values than small sized organizations. In the organizations where processes are defined and executed, expert judgment and similarity methods produce better estimation results.

In conclusion, for the best estimation results, it can be recommended that newly founded local software firms should start with estimation methods base on expert judgment since there is not enough past data. At the same time, an organization should keep track of the previously completed software project data in order to use it in the future. As the number of the previously completed software projects increases data of the completed projects is accumulated. The organization should also use one of the algorithmic models and calibrate the model with recorded data. For example, COCOMO is a widely known and good estimation method and there exist a great deal of free COCOMO estimation tools and information that can be found on the Internet. As the estimation practice maturity of the organization increases any estimation should be crosschecked with different estimation methods. It is recommended that the estimation be crosschecked with at least 2 different estimation methods. Finally, organizations should record each estimation performance and try to define methods, procedures or processes to improve it.

REFERENCES

Oxford Advanced Learner's Dictionary. (2005). Oxford: Oxford University Press.

Albrecht, Allan J.;. (1979). Measuring Application Development Productivity. In I. Press (Ed.), *Proc. IBM Application Development Symposium* (pp. 83-92). Monterey, CA: IBM Press.

Alican, F. (2006). *Türkiye Yazılım Stratejisi.* İstanbul: Yazılım Sanayicileri Derneği.

Basha, S., & P., D. (2010). Analysis of Empirical Software Effort Estimation Models. *International Journal of Computer Science and Information Security, 7*(3), 69-70.

Boehm, B. W. (1981). *Software Engineering Economics.* New Jersey: Prentince Hall, Inc.

Dalkey, N., & Helmer, O. (1963, April). An Experimental Application of the Delphi Method to the Use of Experts. *Management Science, 9*(3), 458-467.

Dalkey, Norman C.;. (1969). *The Delphi Method: An Experimental Study on Group Opinion.* Santa Monica, CA, USA: RAND Corporation.

DPT. (2007). *Bilgi ve İletişim Teknolojileri Özel İhtisas Komisyonu Raporu.* Ankara: DPT.

Güzel, M. (2009, May 20). Estimation in IT Projects. *Estimation in IT Projects*. İstanbul, Turkey: İstanbul Bilgi University.

Hennessy, J. (2004, September 1). *ISD Wideband Delphi Estimation.* Retrieved March 23, 2010, from NASA Goddard Space Flight Center Software Process Improvement: http://software.gsfc.nasa.gov/assetsapproved/PA1.2.1.2.pdf

Hihn, J., & Habib-Agahi, H. (1991). Cost estimation of software intensive projects: A survey of current. *International Conference on Software Engineering, IEEE Comput. Soc. Press* (pp. 276-287). Los Alamitos, CA, USA: IEEE Comput. Soc. Press.

Jeffery, R. (1987). Time-Sensitive Cost Models in the Commercial MIS Environment. *IEEE Transactions on Software Engineering*, 852-859.

Jørgensen, M. (2002). *A Review of Studies on Expert Estimation of Software Development Effort.* Lysaker, Norway: Simula Research Laboratory,.

Karner, G. (1993, December 21). Metrics for Objectory. *Diploma thesis*. Linköping, Sweden: University of Linköping.

Kitchenham, B. A. (1992). Empirical Studies of Assumptions That Underlie Software Cost-Estimation Models. *Information and Software Technology*, 211-218.

Kitchenham, B. A., & Taylor, N. R. (1984). Software Cost Models. *ICL Technical Journal*, 73-102.

Kitchenham, B. A., & Taylor, N. R. (1984). Software Cost Models. *ICL Technical Journal*, 73-102.

Kitchenham, B., Pfleeger, S. L., McColl, B., & Eagan, S. (2002). A Case Study of Maintenance Estimation Accuracy. *Journal of Systems and Software*.

Laird, L. M., & Brennan, M. C. (2006). *Software Measurement and Estimation: A Practical Approach.* Hoboken, New Jersey: A John Wiley & Sons, Inc.

Linstone, H. A., & Turoff, M. (2002). *The Delphi Method, Techniques and Applications.* Newark, NJ: New Jersey Institute of Technology.

McConnell, S. (2006). *Software Estimation, Demystifying the Black Art.* Redmond, Washington: Microsoft Press.

McGibbon, T. (1997, August 20). *Modern Cost Estimation Tools and Modern Emprical Cost and Schedule Estimation Tools.* Retrieved March 10, 2010, from Data & Analysis Center for Software: http://www.dacs.dtic.mil/techs/estimation/title.shtml

Parr, F. N. (1980). An Alternative to the Rayleigh Curve Model for Software Development Effort. *IEEE Transactions on Software Engineering*, 291 - 296.

Paynter, J. (1996). Project Estimation Using Screenflow Engineering. *International Conference on Software Engineering: Education and Practice Dunedin, New Zealand* (pp. 150-159). Los Alamitos, CA, USA: IEEE Computer Society Press.

Rowe, G., & Wright, G. (2001). *Expert opinions in forecasting: The role of the Delphi Technique.* Boston: Kluwer Academic.

Shepperd, M., Schofield, C., & Kitchenham, B. (1996). Effort Estimation Using Analogy. *18th International Conference on Software Engineering* (pp. 170-178). Washington DC, USA: IEEE Computer Society.

Symons, C. R. (1988, January). Function Point Analysis: Difficulties and Improvements. *IEEE Transactions on Software Engineering, 14*(1), 2-11.

Walkerden, F., & Jeffery, R. (1999, June). An Empirical Study of Analogy-based Software Effort Estimation. *Empirical Software Engineering, 4*(2), 135-158.

Wolverton, R. W. (1974, June). The Cost of Developing Large-Scale Software. *IEEE Transactions, Computers*, pp. 615-636.

APPENDIX A DEFINITIONS

Table A-1 Definitions

Term	Definition
Software Size	Software size is a measure used as an input in estimation process of development activity cost. Unit of software size can be in lines of code, function points, story points, etc.
Expert	A person who has knowledge about both business and technical domains and some development experience on technical domain
Estimation	Prediction of the required effort to fulfill certain task
Gearing Factor	Relative LOC required to be coded with different programming languages
Transaction	An event which occurs entirely or not at all.

APPENDIX B LANGUAGE GEARING FACTORS

Table B-1 Language Gearing Factors

	QSM SLOC / FP Data				
Language	Average	Median	Low	High	David Consulting Group Data
Access	35	38	15	47	—
Ada	154	—	104	205	—
Advantage	38	38	38	38	—
APS	86	83	20	184	—
ASP	69	62	32	127	—
Assembler	172	157	86	320	575 Basic/400 Macro
C	148	104	9	704	225
C++	60	53	29	178	80
C#	59	59	51	66	—
Clipper	38	39	27	70	60
Cobol	73	77	8	400	175
Cool:Gen/IEF	38	31	10	180	—
Culprit	51	—	—	—	—
Dbase III	—	—	—	—	60
Dbase IV	52	—	—	—	55
Easytrieve +	33	34	25	41	—
Excel	47	46	31	63	—
Focus	43	42	32	56	60

FORTRAN	—	—	—	—	210
FoxPro	32	35	25	35	—
HTML	43	42	35	53	
Ideal	66	52	34	203	—
IEF/Cool:Gen	38	31	10	180	—
Informix	42	31	24	57	—
J2EE	61	50	40	60	—
Java	60	59	14	97	80
JavaScript	56	54	44	65	50
JCL	60	48	21	115	400
JSP	59	—	—	—	—
Lotus Notes	21	22	15	25	—
Mantis	71	27	22	250	—
Mapper	118	81	16	245	—
Natural	60	52	22	141	100
Oracle	38	29	4	122	60
Oracle Dev 2K/FORMS	41/42	30	21/23	100	—
Pacbase	44	48	26	60	—
PeopleSoft	33	32	30	40	—
Perl	60	—	—	—	50
PL/1	59	58	22	92	126
PL/SQL	46	31	14	110	—
PowerBuilder	30	24	7	105	—

REXX	67	—	—	—	—
RPG II/III	61	49	24	155	120
Sabretalk	80	89	54	99	—
SAS	40	41	33	49	50
Siebel Tools	13	13	5	20	—
Slogan	81	82	66	100	—
Smalltalk	35	32	17	55	—
SQL	39	35	15	143	—
VBScript	45	34	27	50	50
Visual Basic	50	42	14	276	—
VPF	96	95	92	101	—
Web Scripts	44	15	9	114	—

(Source: Laird & Brennan, 2006, pp. 38-39)

APPENDIX C SOFTWARE DEVELOPMENT ACTIVITIES COMMONLY MISSING FROM SOFTWARE ESTIMATES

Table C-1 Commonly Missing Activities from Software Estimates

Activity
Ramp-up time for new team members
Mentoring of new team members
Management coordination/manager meetings
Cutover/deployment
Data conversion
Installation
Customization
Requirements clarifications
Maintaining the revision control system
Supporting the build
Maintaining the scripts required to run the daily build
Maintaining the automated smoke test used in conjunction with the daily build
Installation of test builds at user location(s)
Creation of test data
Management of beta test program
Participation in technical reviews
Integration work
Processing change requests
Attendance at change-control/triage meetings
Coordinating with subcontractors
Technical support of existing systems during the project
Maintenance work on previous systems during the project

Defect-correction work
Performance tuning
Learning new development tools
Administrative work related to defect tracking
Coordination with test (for developers)
Coordination with developers (for test)
Answering questions from quality assurance
Input to user documentation and review of user documentation
Review of technical documentation
Demonstrating software to customers or users
Demonstrating software at trade shows
Demonstrating the software or prototypes
of the software to upper management, clients, and end users
Interacting with clients or end users; supporting beta installations at client locations
Reviewing plans, estimates, architecture, detailed designs, stage plans, code, test cases, and so on

(Source: McConnell, 2006, p. 45)

APPENDIX D CATEGORICAL QUESTION ANSWERS AND AVERAGE OF LOGARITHM OF EEA VALUES

Table D-1 Average of Logarithm of EEA Values With Respect to Firms Size

Question	Answer	Number of Answers	Average of Log(EEA)	Estimate for Difference	T-Value	P-Value
What is the average number of employees working in your organization since June 2009?	<= 50	8	0,71	0,267	1,45	0,178
	> 50 And < 250	18	0,45			

Table D-2 Average of Logarithm of EEA Values with Respect to Internationality of Firm

Question	Answer	Number of Answers	Average of Log (EEA)
What is the classification of your organization with respect to being national or international?	National	17	0,62
	International	9	0,36
	Multinational	0	-

Table D-3 Average of Logarithm of EEA Values with Respect to Number of Owners

Question	Answer	Number of Answers	Average of Log (EEA)
If your organization is an organization of an association, how many associates does it have?	Owned by one employer	1	0,16
	Shares are available to public	7	0,47
	Number of associates <= 8	18	0,57

Table D-4 Average of Logarithm of EEA Values with Respect to Ownership

Question	Answer	Number of Answers	Average of Log (EEA)
If your organization is an organization of an association, what is the classification of your organization with respect to its association structure?	Owned by one employer	1	0,16
	Incorporation	20	0,47
	General Partnership	5	0,83

Table D-5 Average of Logarithm of EEA Values with Respect to System Requirement Analysis Process Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is system requirement analysis process defined in your organization?	Yes	12	0,60	-0,126	-0,79	0,441
	No	14	0,47			

Table D-6 Average of Logarithm of EEA Values with Respect to Integration Process Definition

Question	Answer	Number of Answers	Average of Log(EEA)	Estimate for Difference	T-Value	P-Value
Are system test and integration processes defined in your organization?	Yes	11	0,57	-0,069	-0,42	0,68
	No	15	0,50			

Table D-7 Average of Logarithm of EEA Values with Respect to Sectors That the Firm is Providing Services

Question	Answer	Number of Answers	Average of Log(EEA)
Please select the sectors that your organization provide services	Food, drink and tobacco production sectors	1	0,36
	Textile and textile production sectors	2	1,16
	Main metal and fabrication metal product production sectors	1	1,19
	Electrical and optical hardware production sectors	2	0,77
	Sectors that produce products which are not classified elsewhere	4	0,36
	Construction sector	1	0,36
	Wholesale and retail trade; maintenance of motor vehicle, motorcycle, personal goods and furniture sectors	2	0,74
	Transportation, storage and communications sectors?	1	0,30
	Finance sector	20	0,49
	Real estate, rental and business activities sectors?	2	0,39
	Institutions that operate in the fields of public administration and defense, obligatory social insurance sectors?	1	0,61

Table D-8 Average of Logarithm of EEA Values with Respect to Maintenance Process Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Are system and software maintenance processes defined in your organization?	Yes	13	0,55	-0,045	-0,29	0,775
	No	13	0,51			

Table D-9 Average of Logarithm of EEA Values with Respect to Quality Management Process Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is quality management process defined in your organization?	Yes	18	0,44	0,284	1,57	0,148
	No	8	0,73			

Table D-10 Average of Logarithm of EEA Values with Respect to Risk Management Process Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is risk management process defined in your organization?	Yes	18	0,45	0,244	1,28	0,231
	No	8	0,70			

Table D-11 Average of Logarithm of EEA Values with Respect to Subcontractor Management Process Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is subcontractor management process defined in your organization?	Yes	8	0,43	0,140	0,85	0,408
	No	18	0,57			

Table D-12 Average of Logarithm of EEA Values with Respect to Project Management Process Maturity

Question	Answer	Number of Answers	Average of Log(EEA)
Which one below describes best the current state of the project management process in your organization?	We have process which is sustained by personal endeavor	2	1,03
	We have an executable process in a planned manner although it has not become an enterprise standard yet.	14	0,42
	We have both executed and defined process in the enterprise level.	8	0,57
	We have both defined and quantitatively governed process.	1	0,53
	We have defined, quantitatively governed and continuously improved process.	1	0,78

Table D-13 Average of Logarithm of EEA Values with Respect to Requirement Management Process Maturity

Question	Answer	Number of Answers	Average of Log(EEA)
Which one below describes best the current state of the requirement management in your organization?	We have process which is sustained by personal endeavor	1	1,19
	We have an executable process in a planned manner although it has not become an enterprise standard yet.	14	0,44
	We have both executed and defined process in the enterprise level.	10	0,57
	We have both defined and quantitatively governed process.	0	-
	We have defined, quantitatively governed and continuously improved process.	1	0,78

Table D-14 Average of Logarithm of EEA Values with Respect to Design Process Maturity

Question	Answer	Number of Answers	Average of Log(EEA)
Which one below describes best the current state of the software design process in your organization?	We have process which is sustained by personal endeavor	2	0,76
	We have an executable process in a planned manner although it has not become an enterprise standard yet.	14	0,47
	We have both executed and defined process in the enterprise level.	9	0,54
	We have both defined and quantitatively governed process.	0	-
	We have defined, quantitatively governed and continuously improved process.	1	0,78

Table D-15 Average of Logarithm of EEA Values with Respect to Software Development Process Maturity

Question	Answer	Number of Answers	Average of Log (EEA)
Which one below describes best the current state of the software development process in your organization?	We have process which is sustained by personal endeavor	1	1,19
	We have an executable process in a planned manner although it has not become an enterprise standard yet.	15	0,46
	We have both executed and defined process in the enterprise level.	9	0,54
	We have both defined and quantitatively governed process.	0	-
	We have defined, quantitatively governed and continuously improved process.	1	0,78

Table D-16 Average of Logarithm of EEA Values with Respect to Software Test Process Maturity

Question	Answer	Number of Answers	Average of Log (EEA)
Which one below describes best the current state of the software testing process in your organization?	We have process which is sustained by personal endeavor	4	0,81
	We have an executable process in a planned manner although it has not become an enterprise standard yet.	12	0,47
	We have both executed and defined process in the enterprise level.	8	0,44
	We have both defined and quantitatively governed process.	1	0,53
	We have defined, quantitatively governed and continuously improved process.	1	0,78

Table D-17 Average of Logarithm of EEA Values with Respect to Maintenance Process Maturity

Question	Answer	Number of Answers	Average of Log (EEA)
Which one below describes best the current state of the software maintenance process in your organization?	We have process which is sustained by personal endeavor	11	0,61
	We have an executable process in a planned manner although it has not become an enterprise standard yet.	5	0,37
	We have both executed and defined process in the enterprise level.	9	0,49
	We have both defined and quantitatively governed process.	0	-
	We have defined, quantitatively governed and continuously improved process.	1	0,17

Table D-18 Average of Logarithm of EEA Values with Respect to Compliance with CMMI

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Are your development processes based on CMMI or SW/CMM?	Yes	18	0,45	0,244	1,28	0,231
	No	8	0,70			

Table D-19 Average of Logarithm of EEA Values with Respect to Compliance with Other Types of Development Methodology

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Are your development processes based on a development methodology not mentioned here?	Yes	6	0,42	0,141	0,75	0,476
	No	20	0,56			

Table D-20 Average of Logarithm of EEA Values with Respect to Compliance with No Type of Development Methodology

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Are your development processes based on no kind of methodology?	Yes	5	0,81	-0,353	-1,41	0,233
	No	21	0,46			

Table D-21 Average of Logarithm of EEA Values with Respect to Customer / Agreement Manager Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the role of Customer / Agreement Manager defined in your organization?	Yes	9	0,41	0,178	1,06	0,309
	No	17	0,59			

Table D-22 Average of Logarithm of EEA Values with Respect to Requirement Engineer / Analyst Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate fo Difference	T-Value	P-Value
Is the role of Requirement Engineer / Analyst defined in your organization?	Yes	19	0,49	0,150	0,76	0,467
	No	7	0,64			

Table D-23 Average of Logarithm of EEA Values with Respect to Architect Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the role of Architect defined in your organization?	Yes	17	0,46	0,196	1,07	0,307
	No	9	0,66			

Table D-24 Average of Logarithm of EEA Values with Graphical User Interface Designer Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the role of Graphical User Interface Designer defined in your organization?	Yes	10	0,55	-0,036	-0,23	0,817
	No	16	0,52			

Table D-25 Average of Logarithm of EEA Values with Respect to Test Engineer Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the role of Test Engineer defined in your organization?	Yes	18	0,52	0,020	0,10	0,921
	No	8	0,54			

Table D-26 Average of Logarithm of EEA Values with Respect to Quality Analyst / Engineer Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the role of Quality Manager / Engineer defined in your organization?	Yes	17	0,46	0,204	1,15	0,272
	No	9	0,66			

Table D-27 Average of Logarithm of EEA Values with Respect to Configuration Manager Role Definition

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the role of Configuration Manager defined in your organization?	Yes	18	0,44	0,284	1,57	0,148
	No	8	0,73			

Table D-28 Average of Logarithm of EEA Values with Respect to Organizational Structure

Question	Answer	Number of Answers	Average of Log (EEA)
Which one below is correct about your organizational structure of your organization?	There exists a hierarchical structure which is mainly functional and based on area of expertise.	7	0,59
	There exists a structure based on project management which relies on resource usage from resource pools of competence	10	0,60
	There exists a matrix structure between areas of expertise and project teams.	7	0,47
	There exist temporary structures designed with respect to project needs and work acquired.	2	0,20
	None of the above.	0	-

Table D-29 Average of Logarithm of EEA Values with Respect to Measurement of Number of Requirements

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Number of Requirements used in your organization?	Yes	14	0,57	-0,090	-0,57	0,577
	No	12	0,48			

Table D-30 Average of Logarithm of EEA Values with Respect to Measurement of Software Defects

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Software Defects used in your organization?	Yes	17	0,52	0,039	0,21	0,834
	No	9	0,56			

Table D-31 Average of Logarithm of EEA Values with Respect to Measurement of Effort by Activity

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Effort by Activity used in your organization?	Yes	14	0,58	-0,107	-0,67	0,510
	No	12	0,47			

Table D-32 Average of Logarithm of EEA Values with Respect to Measurement of Software Size

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Software Size used in your organization?	Yes	9	0,41	0,187	1,19	0,249
	No	17	0,59			

Table D-33 Average of Logarithm of EEA Values with Respect to Measurement of Staff Estimated / Assigned per Project Phase

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Staff Estimated / Assigned per Project Phase used in your organization?	Yes	8	0,57	-0,064	-0,35	0,731
	No	18	0,51			

Table D-34 Average of Logarithm of EEA Values with Respect to Measurement of Time Estimated / Spent per Project Phase

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Time Estimated / Spent by Project Phase is used in your organization?	Yes	21	0,59	-0,299	-1,88	0,101
	No	5	0,29			

Table D-35 Average of Logarithm of EEA Values with Respect to Measurement of Cost Estimated / Realized per Project Phase

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the software metric of Cost Estimated / Realized per Project Phase is used in organization?	Yes	12	0,62	-0,169	-1,07	0,300
	No	14	0,45			

Table D-36 Average of Logarithm of EEA Values with Respect to Budget Limitations Preventing Software Measurements

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Do the budget limitations prevent applications of software measurements to be adopted?	Yes	6	0,45	0,104	0,50	0,638
	No	20	0,55			

Table D-37 Average of Logarithm of EEA Values with Respect to Insufficient Support from Senior Management Preventing Software Measurements

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Does the insufficient support from senior management prevent applications of software measurements to be adopted?	Yes	19	0,51	0,083	0,40	0,697
	No	7	0,59			

Table D-38 Average of Logarithm of EEA Values with Respect to Reluctance of Employees Preventing Software Measurements

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Does the reluctance of employees to such applications prevent applications of software measurements to be adopted?	Yes	18	0,51	0,064	0,31	0,760
	No	8	0,57			

Table D-39 Average of Logarithm of EEA Values with Respect to Insufficient Technical Information Preventing Software Measurements

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Does the insufficient technical information or difficulties in obtaining it prevent applications of software measurements to be adopted?	Yes	6	0,58	-0,065	-0,35	0,736
	No	20	0,52			

Table D-40 Average of Logarithm of EEA Values with Respect to Inconvenient Organizational Culture Preventing Software Measurements

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Does the inconvenient organizational culture prevent applications of software measurements to be adopted?	Yes	21	0,53	-0,004	-0,02	0,986
	No	5	0,53			

Table D-41 Average of Logarithm of EEA Values with Respect to Usage of Expert Judgment as Estimation Method in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is Expert Judgment Method used as an estimation method in the sample project?	Yes	20	0,51	0,105	0,63	0,542
	No	6	0,61			

Table D-42 Average of Logarithm of EEA Values with Respect to Usage of Similarity as Estimation Method in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is Similarity Method used as an estimation method in the sample project?	Yes	12	0,52	0,027	0,17	0,863
	No	14	0,54			

Table D-43 Average of Logarithm of EEA Values with Respect to Usage of Decomposition of Tasks as Estimation Method in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is Decomposition of Tasks used as an estimation method in the sample project?	Yes	7	0,65	-0,160	-0,85	0,420
	No	19	0,49			

Table D-44 Average of Logarithm of EEA Values with Respect to Usage of Decomposition of System as Estimation Method in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is Decomposition of System used as an estimation method in the sample project?	Yes	8	0,68	-0,217	-1,31	0,215
	No	18	0,46			

Table D-45 Average of Logarithm of EEA Values with Respect to Complexity of the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)
How many weeks does it take in order to adopt a developer who was not a project team member by means of code review?	A developer can understand and adapt to the product in 1 week.	2	1,03
	A developer can understand and adapt to the product in 2 weeks.	8	0,42
	A developer can understand and adapt to the product in 4 weeks.	11	0,51
	A developer can understand and adapt to the product in 8 weeks.	4	0,54
	A developer can understand and adapt to the product more than 8 weeks.	1	0,61

Table D-46 Average of Logarithm of EEA Values with Respect to Documentation of Contract Management in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the documentation of contract management practice realized in the sample project?	Yes	16	0,59	-0,151	-1,03	0,316
	No	10	0,44			

Table D-47 Average of Logarithm of EEA Values with Respect to Documentation of Design Practice in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the documentation of design practice realized in the sample project?	Yes	18	0,57	-0,141	-0,89	0,386
	No	8	0,43			

Table D-48 Average of Logarithm of EEA Values with Respect to Documentation of Test Cases in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the documentation of test cases practice realized in the sample project?	Yes	12	0,48	0,095	0,63	0,535
	No	14	0,57			

Table D-49 Average of Logarithm of EEA Values with Respect to Documentation of Test Results in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is the documentation of test results practice realized in the sample project?	Yes	15	0,41	0,292	1,91	0,074
	No	11	0,70			

Table D-50 Average of Logarithm of EEA Values with Respect to Documentation of Product Manuals in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Are the documentation practices of product help, maintenance and support realized in the sample project?	Yes	16	0,55	-0,041	-0,24	0,815
	No	10	0,51			

Table D-51 Average of Logarithm of EEA Values with Respect to Storage Requirement of the Product

Question	Answer	Number of Answers	Average of Log (EEA)
What is the storage requirement of the product?	Product does not store any data.	1	1,12
	Product stores data lower than 1 megabyte.	0	-
	Product stores data between 1 megabyte and 1 gigabyte.	11	0,46
	Product stores data between 1 gigabyte and 1 terra bytes.	13	0,56
	Product stores data greater than 1 terra bytes.	1	0,33

Table D-52 Average of Logarithm of EEA Values with Respect to Code Reused in Product Development

Question	Answer	Number of Answers	Average of Log (EEA)
What is the percentage of the code reused with respect to code of the product?	There was no line code reused.	2	0,32
	The percentage of the code reused is lower than 25%.	8	0,55
	The percentage of the code reused is between 25% and 50%.	13	0,51
	The percentage of the code reused is between 50% and 75%.	3	0,70
	The percentage of the code reused is greater than 75%.	0	-

Table D-53 Average of Logarithm of EEA Values with Respect to Strategic Importance of the Product

Question	Answer	Number of Answers	Average of Log(EEA)
If strategic importance of the sample project in your organization is needed to be assessed, please select the appropriate choice.	It has no effect on achievement of the organizations' mission.	0	-
	It has a little effect on achievement of the organizations' mission.	3	0,60
	It has some effect on achievement of the organizations' mission.	12	0,49
	It has a lot of effect on achievement of the organizations' mission.	9	0,60
	This project was the prime target of the organization.	2	0,36

Table D-54 Average of Logarithm of EEA Values with Respect to Reliability Requirement of the Product

Question	Answer	Number of Answers	Average of Log(EEA)
Please select the reliability requirement of software product produced at the end of the sample project?	< 50 %	2	0,98
	50 % - 80 %	6	0,70
	80 % - 90 %	3	0,43
	90 % - 95 %	3	0,50
	> 95 %	12	0,40

Table D-55 Average of Logarithm of EEA Values with Respect to Availability Requirement of the Product

Question	Answer	Number of Answers	Average of Log(EEA)
Please select the availability requirement of software product produced at the end of the sample project?	< 95 %	2	0,33
	95 % - 99 %	3	0,70
	99 % - 99.9 %	2	0,54
	99 % - 99.9 %	10	0,58
	> 99.99 %	9	0,46

Table D-56 Average of Logarithm of EEA Values with Respect to Usage of Java Programming Language in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is Java programming language used in sample project?	Yes	13	0,64	-0,212	-1,42	0,170
	No	13	0,42			

Table D-57 Average of Logarithm of EEA Values with Respect to Usage of C# Programming Language in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is C# programming language used in sample project?	Yes	17	0,47	0,173	1,00	0,333
	No	9	0,64			

Table D-58 Average of Logarithm of EEA Values with Respect to Usage of COBOL Programming Language in the Sample Project

Question	Answer	Number of Answers	Average o f Log (EEA)	Estimate for Difference	T-Value	P-Value
Is COBOL programming language used in sample project?	Yes	9	0,45	0,127	0,81	0,430
	No	17	0,57			

Table D-59 Average of Logarithm of EEA Values with Respect to Usage of Other Programming Languages in the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)	Estimate for Difference	T-Value	P-Value
Is another programming language not mentioned above used in sample project?	Yes	7	0,61	-0,103	-0,49	0,635
	No	19	0,50			

Table D-60 Average of Logarithm of EEA Values with Respect to Project Team Capability

Question	Answer	Number of Answers	Average of Log (EEA)
If average capability of the project team is need to be assessed, please select the appropriate choice.	Majority of the project team is made up of incompetent staffs.	2	0,55
	Majority of the project team is made up of less competent staffs.	2	0,46
	Majority of the project team is made up of competent staffs.	18	0,44
	Majority of the project team is made up of highly competent staffs.	2	1,13
	The project team is completely made up of highly competent staffs.	2	0,76

Table D-61 Average of Logarithm of EEA Values with Respect to Average of Percentage of Effort Spent by Project Team for the Sample Project

Question	Answer	Number of Answers	Average of Log (EEA)
If percentage of effort spent by the project team for the sample project is need to be assessed, please select the appropriate choice.	The project team spent 20 % of their total effort for the sample project tasks.	4	0,77
	The project team spent 20 % to 40 % of their total effort for the sample project tasks.	1	0,30
	The project team spent 40 % to 60 % of their total effort for the sample project tasks.	9	0,63
	The project team spent 60 % to 80 % of their total effort for the sample project tasks.	7	0,39
	The project team spent more than 80 % of their total effort for the sample project tasks.	5	0,39

Table D-62 Average of Logarithm of EEA Values with Respect to Experience of Project Team on the Platforms Used in the Sample Project

Question	Answer	Number of Answers	Average of Log(EEA)
If average experience of the project team on software platforms used in the sample project is need to be assessed, please select the appropriate choice.	Majority of the project team does not have any experience on the platforms used in the sample project.	1	0,61
	Majority of the project team has used the platforms used in the sample project less than 5 projects.	3	0,68
	Majority of the project team has used the platforms used in the sample project 5 up to 10 projects.	10	0,52
	Majority of the project team has used the platforms used in the sample project 10 up to 20 projects.	11	0,43
	Majority of the project team has used the platforms used in the sample project more than 20 projects.	1	1,19

Table D-63 Average of Logarithm of EEA Values with Respect to Experience of Project Team on the Application Type Developed in the Sample Project

Question	Answer	Number of Answers	Average of Log(EEA)
If average experience of the project team about the type of the application developed in the sample project is need to be assessed, please select the appropriate choice.	Majority of the project team does not have experience on type of the application.	7	0,41
	Majority of the project team has developed less than 5 application of this type.	3	0,62
	Majority of the project team has developed 5 up to 10 application of this type.	6	0,57
	Majority of the project team has developed 10 up to 20 application of this type.	9	0,50
	Majority of the project team has developed more than 20 application of this type.	1	1,19

Table D-64 Average of Logarithm of EEA Values with Respect to Experience of Project Team on the Programming Languages Used in the Sample Project

Question	Answer	Number of Answers	Average of Log(EEA)
If average experience of the project team about the programming languages used in the sample project is need to be assessed, please select the appropriate choice.	Majority of the project team does not have any experience on the programming languages used in the sample project.	0	-
	Majority of the project team has used the programming languages used in the sample project less than 5 projects.	3	0,46
	Majority of the project team has used the programming languages used in the sample project 5 up to 10 projects.	7	0,52
	Majority of the project team has used the programming languages used in the sample project 10 up to 20 projects.	14	0,45
	Majority of the project team has used the programming languages used in the sample more than 20 projects.	2	1,24

Table D-65 Average of Logarithm of EEA Values with Respect to Memory Limitation in Production Environment

Question	Answer	Number of Answers	Average of Log(EEA)
Is there any memory limitation in production environment?	Software product is allowed to use very limited memory resources in the production environment.	2	0,26
	Software product is allowed to use limited memory resources in the production environment that cannot be consumed easily.	8	0,36
	Software product is allowed to use unlimited memory resources in the production environment.	16	0,65

Table D-66 Average of Logarithm of EEA Values with Respect to CPU Time Limitation in Production Environment

Question	Answer	Number of Answers	Average of Log(EEA)
Is there any CPU time limitation in production environment?	Software product is allowed to use very limited CPU time in the production environment.	2	0,26
	Software product is allowed to use limited CPU time in the production environment that cannot be spent easily.	7	-0,29
	Software product is allowed to use unlimited CPU time in the production environment.	17	0,66

Table D-67 Average of Logarithm of EEA Values with Respect to Communication Bandwidth Limitation in Production Environment

Question	Answer	Number of Answers	Average of Log(EEA)
Is there any communication bandwidth limitation in production environment?	Software product is allowed to use very limited range of the network bandwidth in the production environment.	1	0.36
	Software product is allowed to use limited range of the network bandwidth in the production environment that cannot be consumed easily.	11	0,42
	Software product is allowed to use unlimited network bandwidth in the production environment.	14	0,63

Table D-68 Average of Logarithm of EEA Values with Respect to Storage Capacity Limitation in Production Environment

Question	Answer	Number of Answers	Average of Log(EEA)
Is there any storage capacity limitation in production environment?	Software product is allowed to use very limited storage capacity in the production environment.	2	0,26
	Software product is allowed to use limited storage capacity in the production environment that cannot be consumed easily.	5	0,50
	Software product is allowed to use unlimited storage capacity in the production environment.	19	0,57

Table D-69 Average of Logarithm of EEA Values with Respect to Relative Size of the Sample Project

Question	Answer	Number of Answers	Average of Log(EEA)
If the size of the project is needed to be assessed with respect to other project completed in your organization, please select appropriate choice below.	It was a very small sized project.	0	-
	It was a small sized project.	7	0,50
	It was a mid-sized project.	12	0,55
	It was a large sized project.	3	0,32
	It was the one of the three largest projects.	4	0,68

Table D-70 Average of Logarithm of EEA Values with Respect to Schedule Limitations of the Sample Project

Question	Answer	Number of Answers	Average of Log(EEA)
If the schedule limitations of the project are needed to be assessed, please select appropriate choice below.	There was no time limitation. Project schedule was extended as needed.	2	0,62
	It was allowed to extend project schedule 50 % of the total project time maximum.	1	0,16
	It was allowed to extend project schedule 20 % of the total project time maximum.	16	0,62
	It was allowed to extend project schedule 5 % of the total project time maximum.	5	0,36
	Project had to be completed on time. Schedule slippage is not allowed.	2	0,32

Table D-71 Average of Logarithm of EEA Values with Respect to Arrival Mean of the Sample Project

Question	Answer	Number of Answers	Average of Log(EEA)
By what means did the sample project arrived to your organization?	Project was started by means of tender won by our organization.	0	-
	Project was started with our customer request directly sent to us.	15	0,59
	Project was started by our organization in order to re-write our COTS.	2	0,11
	Project was started by our organization.	1	0,78
	Proje was brought to our organization by third party.	8	0,48

APPENDIX E CROSS TABULAR STATISTICS OF SURVEY QUESTIONS

Table E-1 Cross Tabular Statistic for Internationality of the Firm and Firm Size

What is the classification of your organization with respect to being national or international?	Small Size	Middle Size	All
National	0,77	0,52	0,62
	7	10	17
International	0,36	0,36	0,36
	1	8	9
All	0,71	0,45	0,53
	8	18	26

Table E-2 Cross Tabular Statistic for Classification of the Ownership Structure of the Firm and Firm Size

What is the classification of your organization with respect to ownership?	Small Size	Middle Size	All
Owned by one employer	0,16	-	0,16
	1	0	1
Incorporation	0,70	0,45	0,47
	2	18	20
General Partnership	0,83	-	0,83
	5	0	5
All	0,71	0,45	0,53
	8	18	26

Table E-3 Cross Tabular Statistic for Finance Sector Providers and Firm Size

Does your organization provide services to finance sector?	Small Size	Middle Size	All
No	0,79	0,03	0,66
	5	1	6
Yes	0,59	0,47	0,49
	3	17	20
All	0,71	0,45	0,53
	8	18	26

Table E-4 Cross Tabular Statistic for Requirement Analysis Process Definition and Firm Size

Is system requirement analysis process defined in your organization?	Small Sized	Middle Sized	All
No	0,62	0,43	0,47
	3	11	14
Yes	0,77	0,47	0,60
	5	7	12
All	0,71	0,45	0,53
	8	18	26

Table E-5 Cross Tabular Statistic for Design Process Definition and Firm Size

Is software design process defined in your organization?	Small Size	Middle Size	All
No	0,70	0,33	0,62
	4	1	5
Yes	0,73	0,45	0,51
	4	17	21
All	0,71	0,45	0,53
	8	18	26

Table E-6 Cross Tabular Statistic for System Test and Integration Process Definition and Firm Size

Are system test and integration processes defined in your organization?	Small Size	Middle Size	All
No	0,62	0,44	0,50
	5	10	15
Yes	0,88	0,45	0,57
	3	8	11
All	0,71	0,45	0,53
	8	18	26

Table E-7 Cross Tabular Statistic for Software Maintenance Process Definition and Firm Size

Are system and software maintenance processes defined in your organization?	Small Size	Middle Size	All
No	0,55	0,49	0,51
	4	9	13
Yes	0,88	0,41	0,55
	4	9	13
All	0,71	0,45	0,53
	8	18	26

Table E-8 Cross Tabular Statistic for Quality Management Process Definition and Firm Size

Is quality management process defined in your organization?	Small Size	Middle Size	All
No	0,91	0,18	0,73
	6	2	8
Yes	0,13	0,48	0,44
	2	16	18
All	0,71	0,45	0,53
	8	18	26

Table E-9 Cross Tabular Statistic for Risk Management Process Definition and Firm Size

Is risk management process defined in your organization?	Small Size	Middle Size	All
No	0,79	0,03	0,70
	7	1	8
Yes	0,16	0,47	0,45
	1	17	18
All	0,71	0,45	0,53
	8	18	26

Table E-10 Cross Tabular Statistic for Subcontractor Management Process Definition and Firm Size

Is subcontractor management process defined in your organization?	Small Size	Middle Size	All
No	0,79	0,43	0,57
	7	11	18
Yes	0,16	0,47	0,43
	1	7	8
All	0,71	0,45	0,53
	8	18	26

Table E-11 Cross Tabular Statistic for Project Management Maturity Level and Firm Size

Which one below describes best the current state of the project management process in your organization?	Small Size	Middle Size	All
We have process which is sustained by personal endeavor.	1,03	-	1,03
	2	0	2
We have an executable process in a planned manner although it has not become an enterprise standard yet.	0,36	0,43	0,42
	3	11	14
We have both executed and defined process in the enterprise level.	0,86	0,40	0,57
	3	5	8
We have both executed and defined process in the enterprise level.	-	0,53	0,53
	0	1	1
We have defined, quantitatively governed and continuously improved process.	-	0,78	0,78
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-12 Cross Tabular Statistic for Requirement Management Maturity Level and Firm Size

Which one below describes best the current state of the requirement management in your organization?	Small Size	Middle Size	All
We have process which is sustained by personal endeavor.	1,19	-	1,19
	1	0	1
We have an executable process in a planned manner although it has not become an enterprise standard yet.	0,45	0,43	0,44
	3	11	14
We have both executed and defined process in the enterprise level.	0,79	0,42	0,57
	4	6	10
We have defined, quantitatively governed and continuously improved process	-	0,78	0,78
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-13 Cross Tabular Statistic for Design Process Maturity Level and Firm Size

Which one below describes best the current state of the software design process in your organization?	Small Size	Middle Size	All
We have process which is sustained by personal endeavor	1,19	0,33	0,76
	1	1	2
We have an executable process in a planned manner although it has not become an enterprise standard yet.	0,55	0,44	0,47
	4	10	14
We have both executed and defined process in the enterprise level.	0,78	0,42	0,54
	3	6	9
We have defined, quantitatively governed and continuously improved process.	-	0,78	0,78
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-14 Cross Tabular Statistic for Software Development Process Maturity Level and Firm Size

Which one below describes best the current state of the software development process in your organization?	Small Size	Middle Size	All
We have process which is sustained by personal endeavor	1,19	-	1,1875
	1	0	1
We have an executable process in a planned manner although it has not become an enterprise standard yet.	0,55	0,43	0,4641
	4	11	15
We have both executed and defined process in the enterprise level.	0,78	0,42	0,5393
	3	6	9
We have both executed and defined process in the enterprise level.	-	0,78	0,7782
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-15 Cross Tabular Statistic for Software Testing Process Maturity Level and Firm Size

Which one below describes best the current state of the software testing process in your organization?	Small Size	Middle Size	All
We have process which is sustained by personal endeavor.	1,06	0,03	0,81
	3	1	4
We have an executable process in a planned manner although it has not become an enterprise standard yet.	0,48	0,47	0,47
	2	10	12
We have both executed and defined process in the enterprise level.	0,52	0,40	0,44
	3	5	8
We have both defined and quantitatively governed process.	-	0,53	0,53
	0	1	1
We have defined, quantitatively governed and continuously improved process.	-	0,78	0,78
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-16 Cross Tabular Statistic for Software Maintenance Process Maturity Level and Firm Size

Which one below describes best the current state of the software maintenance process in your organization?	Small Size	Middle Size	All
We have process which is sustained by personal endeavor	1,16	0,49	0,61
	2	9	11
We have an executable process in a planned manner although it has not become an enterprise standard yet.	0,48	0,30	0,37
	2	3	5
We have both executed and defined process in the enterprise level.	0,61	0,40	0,49
	4	5	9
We have defined, quantitatively governed and continuously improved process.	-	0,78	0,78
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-17 Cross Tabular Statistic for Selection of CMMI Model and Firm Size

Are your development processes based on CMMI or SW/CMM?	Small Size	Middle Size	All
No	0,79	0,03	0,70
	7	1	8
Yes	0,16	0,47	0,45
	1	17	18
All	0,71	0,45	0,53
	8	18	26

Table E-18 Cross Tabular Statistic for Selection of Other Types of Reference Model and Firm Size

Are your development processes based on a development methodology not mentioned here?	Small Size	Middle Size	All
No	0,71	0,46	0,56
	8	12	20
Yes	-	0,42	0,42
	0	6	6
All	0,71	0,45	0,53
	8	18	26

Table E-19 Cross Tabular Statistic for Selection of No Type of Reference Model and Firm Size

Are your development processes based on no kind of methodology?	Small Size	Middle Size	All
No	0,55	0,45	0,46
	3	18	21
Yes	0,81	-	0,81
	5	0	5
All	0,71	0,45	0,53
	8	18	26

Cross Tabular Statistic for Definition of Customer / Agreement Manager Role and Firm Size

Is the role of Customer/Agreement Manager defined in your organization?	Small Size	Middle Size	All
Yes	0,86	0,48	0,59
	5	12	17
No	0,46	0,39	0,41
	3	6	9
All	0,71	0,45	0,53
	8	18	26

Table E-20 Cross Tabular Statistic for Definition of Requirement Engineer / Analyst Role and Firm Size

Is the role of Requirement Engineer/Analyst defined in your organization?	Small Size	Middle Size	All
No	0,73	0,41	0,64
	5	2	7
Yes	0,68	0,45	0,49
	3	16	19
All	0,71	0,45	0,53
	8	18	26

Table E-21 Cross Tabular Statistic for Definition of Architect Role and Firm Size

Is the role of Architect defined in your organization?	Small Size	Middle Size	All
No	0,73	0,41	0,66
	7	2	9
Yes	0,61	0,45	0,46
	1	16	17
All	0,71	0,45	0,53
	8	18	26

Table E-22 Cross Tabular Statistic for Graphical User Interface Designer Role and Firm Size

Is the role of Graphical User Interface Designer defined in your organization?	Small Size	Middle Size	All
No	0,65	0,41	0,52
	7	9	16
Yes	1,12	0,49	0,56
	1	9	10
All	0,71	0,45	0,54
	8	18	26

Table E-23 Cross Tabular Statistic for Definition of Test Engineer Role and Firm Size

Is the role of Test Engineer defined in your organization?	Small Size	Middle Size	All
No	0,66	0,18	0,54
	6	2	8
Yes	0,87	0,48	0,52
	2	16	18
All	0,71	0,45	0,53
	8	18	26

Table E-24 Cross Tabular Statistic for Definition of Quality Engineer / Manager Role and Firm Size

Is the role of Quality Manager/Engineer defined in your organization?	Small Size	Middle Size	All
No	0,80	0,18	0,66
	7	2	9
Yes	0,11	0,48	0,46
	1	16	17
All	0,71	0,45	0,53
	8	18	26

Table E-25 Cross Tabular Statistic for Definition of Configuration Manager Role and Firm Size

Is the role of Configuration Manager defined in your organization?	Small Size	Middle Size	All
No	0,91	0,18	0,73
	6	2	8
Yes	0,13	0,48	0,44
	2	16	18
All	0,71	0,45	0,53
	8	18	26

Table E-26 Cross Tabular Statistic for Organizational Structure and Firm Size

Which one below is correct about your organizational structure of your organization?	Small Size	Middle Size	All
There exists a hierarchical structure which is mainly functional and based on area of expertise.	0,61	0,55	0,59
	5	2	7
There exists a structure based on project management which relies on resource usage from resource pools of competence.	1,16	0,46	0,60
	2	8	10
There exists a matrix structure between areas of expertise and project teams	-	0,47	0,47
	0	7	7
There exist temporary structures designed with respect to project needs and work acquired.	0,36	0,03	0,20
	1	1	2
All	0,71	0,45	0,53
	8	18	26

Table E-27 Cross Tabular Statistic for Usage of Number of Requirement Metric and Firm Size

Is the software metric of Number of Requirements used in your organization?	Small Size	Middle Size	All
No	0,72	0,36	0,48
	4	8	12
Yes	0,71	0,52	0,57
	4	10	14
All	0,71	0,45	0,53
	8	18	2

Table E-28 Cross Tabular Statistic for Usage of Software Defects Metric and Firm Size

Is the software metric of Software Defects used in your organization?	Small Size	Middle Size	All
No	0,82	0,42	0,56
	3	6	9
Yes	0,65	0,46	0,52
	5	12	17
All	0,71	0,45	0,53
	8	18	26

Table E-29 Cross Tabular Statistic for Usage of Effort by Activity Metric and Firm Size

Is the software metric of Effort by Activity used in your organization?	Small Size	Middle Size	All
No	0,62	0,37	0,47
	5	7	12
Yes	0,87	0,50	0,58
	3	11	14
All	0,71	0,45	0,53
	8	18	26

Table E-30 Cross Tabular Statistic for Usage of Software Size Metric and Firm Size

Is the software metric of Software Size used in your organization?	Small Size	Middle Size	All
No	0,71	0,49	0,59
	8	9	17
Yes	-	0,41	0,41
	0	9	9
All	0,71	0,45	0,53
	8	18	26

Table E-31 Cross Tabular Statistic for Usage of Staff Estimated / Assigned per Project Phase Metric and Firm Size

Is the software metric of Staff Estimated/Assigned per Project Phase used in your organization?	Small Size	Middle Size	All
No	0,61	0,46	0,51
	6	12	18
Yes	1,03	0,42	0,57
	2	6	8
All	0,71	0,45	0,53
	8	18	26

Table E-32 Cross Tabular Statistic for Usage of Time Estimated / Assigned per Project Phase Metric and Firm Size

Is the software metric of Time Estimated/Spent by Project Phase is used in your organization?	Small Size	Middle Size	All
No	0,21	0,41	0,29
	3	2	5
Yes	1,02	0,45	0,59
	5	16	21
All	0,71	0,45	0,53
	8	18	26

Table E-33 Cross Tabular Statistic for Usage of Cost Estimated / Realized per Project Phase Metric and Firm Size

Is the software metric of Cost Estimated/Realized per Project Phase is used in organization?	Small Size	Middle Size	All
No	0,37	0,47	0,45
	3	11	14
Yes	0,92	0,41	0,62
	5	7	12
All	0,71	0,45	0,53
	8	18	26

Table E-34 Cross Tabular Statistic for Budget Limitations Preventing Software Measurements to be Adopted and Firm Size

Do the budget limitations prevent applications of software measurements to be adopted?	Small Size	Middle Size	All
No	0,85	0,48	0,55
	4	16	20
Yes	0,58	0,18	0,45
	4	2	6
All	0,71	0,45	0,53
	8	18	26

Table E-35 Cross Tabular Statistic for Insufficient Support from Senior Management Preventing Software Measurements to be Adopted and Firm Size

Does the insufficient support from senior management prevent applications of software measurements to be adopted?	Small Size	Middle Size	All
No	0,75	0,38	0,59
	4	3	7
Yes	0,68	0,46	0,51
	4	15	19
All	0,71	0,45	0,53
	8	18	26

Table E-36 Cross Tabular Statistic for Reluctance of Employees Preventing Software Measurements to be Adopted and Firm Size

Does the reluctance of employees to such applications prevent applications of software measurements to be adopted?	**Small Size**	**Middle Size**	**All**
No	0,71	0,18	0,57
	6	2	8
Yes	0,74	0,48	0,51
	2	16	18
All	0,71	0,45	0,53
	8	18	26

Table E-37 Cross Tabular Statistic for Insufficient Technical Information Preventing Software Measurements to be Adopted and Firm Size

Does the insufficient technical information or difficulties in obtaining it prevent applications of software measurements to be adopted?	**Small Size**	**Middle Size**	**All**
No	0,86	0,45	0,52
	3	17	20
Yes	0,63	0,33	0,58
	5	1	6
All	0,71	0,45	0,53
	8	18	26

Table E-38 Cross Tabular Statistic for Inconvenient Organizational Culture Preventing Software Measurements to be Adopted and Firm Size

Does the inconvenient organizational culture prevent applications of software measurements to be adopted?	Small Size	Middle Size	All
No	0,52	0,54	0,53
	3	2	5
Yes	0,83	0,44	0,53
	5	16	21
All	0,71	0,45	0,53
	8	18	26

Table E-39 Cross Tabular Statistic for Usage of Expert Judgment Estimation Method and Firm Size

Is Expert Judgment Method used as an estimation method in the sample project?	Small Size	Middle Size	All
No	1,00	0,41	0,61
	2	4	6
Yes	0,62	0,46	0,51
	6	14	20
All	0,71	0,45	0,53
	8	18	26

Table E-40 Cross Tabular Statistic for Usage of Similarity Estimation Method and Firm Size

Is Similarity Method used as an estimation method in the sample project?	Small Size	Middle Size	All
No	0,83	0,50	0,54
	2	12	14
Yes	0,68	0,35	0,52
	6	6	12
All	0,71	0,45	0,53
	8	18	26

Table E-41 Cross Tabular Statistic for Usage of Decomposition of Tasks Estimation Method and Firm Size

Is Decomposition of Tasks used as an estimation method in the sample project?	Small Size	Middle Size	All
No	0,76	0,41	0,49
	4	15	19
Yes	0,67	0,62	0,65
	4	3	7
All	0,71	0,45	0,53
	8	18	26

Table E-42 Cross Tabular Statistic for Usage of Decomposition of System Estimation Method and Firm Size

Is Decomposition of the System used as an estimation method in the sample project?	Small Size	Middle Size	All
No	0,57	0,44	0,46
	3	15	18
Yes	0,80	0,48	0,68
	5	3	8
All	0,71	0,45	0,53
	8	18	26

Table E-43 Cross Tabular Statistic for Product Complexity and Firm Size

How many weeks does it take in order to adopt a developer by means of code review?	Small Size	Middle Size	All
A developer can understand and adapt to the product in 1 week.	1,03	-	1,03
	2	0	2
A developer can understand and adapt to the product in 2 weeks	0,44	0,40	0,42
	4	4	8
A developer can understand and adapt to the product in 4 weeks.	1,29	0,43	0,51
	1	10	11
A developer can understand and adapt to the product in 8 weeks.	-	0,54	0,54
	0	4	4
A developer can understand and adapt to the product more than 8 weeks.	0,61	-	0,61
	1	0	1
All	0,71	0,45	0,53
	8	18	26

Table E-44 Cross Tabular Statistic for Documentation of Contract Management in the Sample Project and Firm Size

Is the documentation of contract management practice realized in the sample project?	Small Size	Middle Size	All
No	0,61	0,36	0,44
	3	7	10
Yes	0,77	0,50	0,59
	5	11	16
All	0,71	0,45	0,53
	8	18	26

Table E-45 Cross Tabular Statistic for Documentation of Design Practice in the Sample Project and Firm Size

Is the documentation of design practice realized in the sample project?	Small Size	Middle Size	All
No	0,77	0,32	0,43
	2	6	8
Yes	0,69	0,51	0,57
	6	12	18
All	0,71	0,45	0,53
	8	18	26

Table E-46 Cross Tabular Statistic for Documentation of Test Cases in the Sample Project and Firm Size

Is the documentation of test cases practice realized in the sample project?	Small Size	Middle Size	All
No	0,79	0,35	0,57
	7	7	14
Yes	0,16	0,51	0,48
	1	11	12
All	0,71	0,45	0,53
	8	18	26

Table E-47 Cross Tabular Statistic for Documentation of Test Results in the Sample Project and Firm Size

Is the documentation of test results practice realized in the sample project?	Small Size	Middle Size	All
No	0,97	0,47	0,70
	5	6	11
Yes	0,29	0,44	0,41
	3	12	15
All	0,71	0,45	0,53
	8	18	26

Table E-48 Cross Tabular Statistic for Documentation of Product Manuals in the Sample Project and Firm Size

Are the documentation product help, maintenance and support practices realized in the sample project?	Small Size	Middle Size	All
No	0,70	0,31	0,51
	5	5	10
Yes	0,74	0,50	0,55
	3	13	16
All	0,71	0,45	0,53
	8	18	26

Table E-49 Cross Tabular Statistic for Storage Requirement of the Product in the Sample Project and Firm Size

What is the storage requirement of the product?	Small Size	Middle Size	All
Product does not store any data.	1,12	-	1,12
	1	0	1
Product stores data between 1 megabyte and 1 gigabyte	0,47	0,46	0,46
	3	8	11
Product stores data between 1 gigabytes and 1 terra byte.	0,80	0,45	0,56
	4	9	13
Product stores data greater than 1 terra byte.	-	0,33	0,33
	0	1	1
All	0,71	0,45	0,53
	8	18	26

Table E-50 Cross Tabular Statistic for Code Reused in the Sample Project and Firm Size

What is the percentage of the code reused with respect to code of the product?	Small Size	Middle Size	All
There was no line code reused.	0,16	0,49	0,32
	1	1	2
The percentage of the code reused is lower than 25%.	0,77	0,47	0,55
	2	6	8
The percentage of the code reused is between 25% and 50%.	0,68	0,44	0,51
	4	9	13
The percentage of the code reused is between 50% and 75%.	1,29	0,41	0,70
	1	2	3
All	0,71	0,45	0,53
	8	18	2

Table E-51 Cross Tabular Statistic for Strategic Importance of the Sample Project and Firm Size

If strategic importance of the sample project in your organization is needed to be assessed, please select the appropriate choice.	Small Size	Middle Size	All
It has a little effect on achievement of the organizations' mission.	0,77	0,25	0,60
	2	1	3
It has some effect on achievement of the organizations' mission.	0,64	0,46	0,49
	2	10	12
It has a lot of effect on achievement of the organizations' mission.	1,08	0,46	0,60
	2	7	9
This project was the prime target of the organization.	0,36	-	0,36
	2	0	2
All	0,71	0,45	0,53
	8	18	26

Table E-52 Cross Tabular Statistic for Reliability Requirement of the Product in the Sample Project and Firm Size

Please select the reliability requirement of software product produced at the end of the sample project?	Small Size	Middle Size	All
< 50 %	1,19	0,78	0,98
	1	1	2
50 % - 80 %	0,86	0,55	0,70
	3	3	6
80 % - 90 %	0,36	0,46	0,43
	1	2	3
90 % - 95 %	-	0,50	0,50
	0	3	3
> 95 %	0,53	0,36	0,40
	3	9	12
All	0,71	0,45	0,53
	8	18	26

Table E-53 Cross Tabular Statistic for Availability Requirement of the Product in the Sample Project and Firm Size

Please select the availability requirement of software product produced at the end of the sample project?	Small Size	Middle Size	All
< 95 %	0,36	0,30	0,33
	1	1	2
95 % - 99 %	0,67	0,75	0,70
	2	1	3
99 % - 99.9 %	0,88	0,20	0,54
	1	1	2
99.9 % - 99.99 %	0,95	0,49	0,58
	2	8	10
> 99.99 %	0,62	0,41	0,46
	2	7	9
All	0,71	0,45	0,53
	8	18	26

Table E-54 Cross Tabular Statistic for Usage of Java Programming Language in the Sample Project and Firm Size

Is Java programming language used in sample project?	Small Size	Middle Size	All
No	0,87	0,34	0,42
	2	11	13
Yes	0,66	0,61	0,64
	6	7	13
All	0,71	0,45	0,53
	8	18	26

Table E-55 Cross Tabular Statistic for Usage of C# Programming Language in the Sample Project and Firm Size

Is C# programming language used in sample project?	Small Size	Middle Size	All
No	0,61	0,67	0,64
	4	5	9
Yes	0,82	0,36	0,47
	4	13	17
All	0,71	0,45	0,53
	8	18	26

Table E-56 Cross Tabular Statistic for Usage of COBOL Programming Language in the Sample Project and Firm Size

Is COBOL programming language used in sample project?	Small Size	Middle Size	All
No	0,71	0,45	0,57
	8	9	17
Yes	-	0,45	0,45
	0	9	9
All	0,71	0,45	0,53
	8	18	26

Table E-57 Cross Tabular Statistic for Usage of Other Programming Language in the Sample Project and Firm Size

Is another programming language not mentioned above used in sample project?	Small Size	Middle Size	All
No	0,55	0,49	0,50
	4	15	19
Yes	0,88	0,24	0,61
	4	3	7
All	0,71	0,45	0,53
	8	18	26

Table E-58 Cross Tabular Statistic for Average Project Capability of Project Team and Firm Size

If average capability of the project team is need to be assessed, please select the appropriate choice.	Small Size	Middle Size	All
Majority of the project team is made up of incompetent staffs.	0,61	0,49	0,55
	1	1	2
Majority of the project team is made up of less competent staffs.	-	0,46	0,46
	0	2	2
Majority of the project team is made up of competent staffs.	0,53	0,41	0,44
	5	13	18
Majority of the project team is made up of highly competent staffs.	1,29	0,96	1,13
	1	1	2
The project team is completely made up of highly competent staffs.	1,19	0,33	0,76
	1	1	2
All	0,71	0,45	0,53
	8	18	26

Table E-59 Cross Tabular Statistic for Percentage of Total Effort Spent by Project Team in the Sample Project and Firm Size

If percentage of effort spent by the project team for the sample project is need to be assessed, please select the appropriate choice.	Small Size	Middle Size	All
The project team spent 20 % of their total effort for the sample project tasks.	0,77	-	0,77
	4	0	4
The project team spent 20 % to 40 % of their total effort for the sample project tasks.	-	0,30	0,30
	0	1	1
The project team spent 40 % to 60 % of their total effort for the sample project tasks.	1,08	0,50	0,63
	2	7	9
The project team spent 60 % to 80 % of their total effort for the sample project tasks.	0,24	0,45	0,39
	2	5	7
The project team spent more than 80 % of their total effort for the sample project tasks.	-	0,39	0,39
	0	5	5
All	0,71	0,45	0,53
	8	18	26

Table E-60 Cross Tabular Statistic for Experience of the Project Team on Platforms in the Sample Project and Firm Size

If average experience of the project team on software platforms used in the sample project is need to be assessed, please select the appropriate choice.	Small Size	Middle Size	All
Majority of the project team does not have any experience on the platforms used in the sample project.	0,61	-	0,61
	1	0	1
Majority of the project team has used the platforms used in the sample project less than 5 projects.	1,00	0,03	0,68
	2	1	3
Majority of the project team has used the platforms used in the sample project 5 up to 10 projects.	0,48	0,55	0,52
	4	6	10
Majority of the project team has used the platforms used in the sample project 10 up to 20 projects.	-	0,43	0,43
	0	11	11
Majority of the project team has used the platforms used in the sample project more than 20 projects.	1,19	-	1,19
	1	0	1
All	0,71	0,45	0,53
	8	18	26

Table E-61 Cross Tabular Statistic for Experience of the Project Team on Application Type in the Sample Project and Firm Size

If average experience of the project team about the type of the application developed in the sample project is need to be assessed, please select the appropriate choice.	**Small Size**	**Middle Size**	**All**
Majority of the project team does not have experience on type of the application.	0,38	0,42	0,41
	2	5	7
Majority of the project team has developed less than 5 application of this type.	0,62	0,61	0,62
	2	1	3
Majority of the project team has developed 5 up to 10 application of this type.	0,62	0,54	0,57
	2	4	6
Majority of the project team has developed 10 up to 20 application of this type.	1,29	0,40	0,50
	1	8	9
Majority of the project team has developed more than 20 application of this type.	1,19	-	1,19
	1	0	1
All	0,71	0,45	0,53
	8	18	26

Table E-62 Cross Tabular Statistic for Experience of the Project Team on Programming Languages in the Sample Project and Firm Size

If average experience of the project team about the programming languages used in the sample project is need to be assessed, please select the appropriate choice.	Small Size	Middle Size	All
Majority of the project team has used the programming languages used in the sample project less than 5 projects.	0,38	0,61	0,46
	2	1	3
Majority of the project team has used the programming languages used in the sample project 5 up to 10 projects.	0,62	0,48	0,52
	2	5	7
Majority of the project team has used the programming languages used in the sample project 10 up to 20 projects.	0,62	0,42	0,45
	2	12	14
Majority of the project team has used the programming languages used in the sample more than 20 projects.	1,24	-	1,24
	2	0	2
All	0,71	0,45	0,53
	8	18	26

Table E-63 Cross Tabular Statistic for Memory Limitations in the Sample Project and Firm Size

Is there any memory limitation in production environment?	Small Size	Middle Size	All
Software product is allowed to use very limited memory resources in the production environment.	0,26	-	0,26
	2	0	2
Software product is allowed to use limited memory resources in the production environment that cannot be consumed easily.	0,11	0,39	0,36
	1	7	8
Software product is allowed to use unlimited memory resources in the production environment.	1,02	0,48	0,65
	5	11	16
All	0,71	0,45	0,53
	8	18	26

Table E-64 Cross Tabular Statistic for CPU Time Limitations in the Sample Project and Firm Size

Is there any CPU time limitation in production environment?	Small Size	Middle Size	All
Software product is allowed to use very limited CPU time in the production environment.	0,26	-	0,26
	2	0	2
Software product is allowed to use limited CPU time in the production environment that cannot be spent easily.	0,11	0,33	0,29
	1	6	7
Software product is allowed to use unlimited CPU time in the production environment.	1,02	0,51	0,66
	5	12	17
All	0,71	0,45	0,53
	8	18	26

Table E-65 Cross Tabular Statistic for Communication Bandwidth Limitations in the Sample Project and Firm Size

Is there any communication bandwidth limitation in production environment?	**Small Size**	**Middle Size**	**All**
Software product is allowed to use very limited range of the network bandwidth in the production environ-ment.	0,36	-	0,36
	1	0	1
Software product is allowed to use limited range of the network bandwidth in the production environment that cannot be consumed easily.	0,52	0,40	0,42
	2	9	11
Software product is allowed to use unlimited network bandwidth in the production environment.	0,86	0,50	0,63
	5	9	14
All	0,71	0,45	0,53
	8	18	26

Table E-66 Cross Tabular Statistic for Storage Capacity Limitations in the Sample Project and Firm Size

Is there any storage capacity limitation in production environment?	Small Size	Middle Size	All
Software product is allowed to use very limited storage capacity in the production environment.	0,26	-	0,26
	2	0	2
Software product is allowed to use limited storage capacity in the production environment that cannot be consumed easily.	-	0,50	0,50
	0	5	5
Software product is allowed to use unlimited storage capacity in the production environment.	0,87	0,43	0,57
	6	13	19
All	0,71	0,45	0,53
	8	18	26

Table E-67 Cross Tabular Statistic for Relative Size of the Sample Project and Firm Size

If the size of the project is needed to be assessed with respect to other project completed in your organization, please select appropriate choice below.	Small Size	Middle Size	All
It was a small sized project.	0,64	0,44	0,50
	2	5	7
It was a mid-sized project.	0,83	0,50	0,55
	2	10	12
It was a large sized project.	0,11	0,43	0,32
	1	2	3
It was the one of the three largest projects.	0,89	0,03	0,68
	3	1	4
All	0,71	0,45	0,53
	8	18	26

Table E-68 Cross Tabular Statistic for Schedule Limitation of the Sample Project and Firm Size

If the schedule limitations of the project are needed to be assessed, please select appropriate choice below.	Small Size	Middle Size	All
There was no time limitation. Project schedule was extended as needed.	0,62	-	0,62
	2	0	2
It was allowed to extend project schedule 50 % of the total project time maximum.	0,16	-	0,16
	1	0	1
It was allowed to extend project schedule 20 % of the total project time maximum.	1,20	0,49	0,62
	3	13	16
It was allowed to extend project schedule 5 % of the total project time maximum.	0,61	0,29	0,36
	1	4	5
Project had to be completed on time. Schedule slippage is not allowed.	0,11	0,53	0,32
	1	1	2
All	0,71	0,45	0,53
	8	18	26

Table E-69 Cross Tabular Statistic for Arrival Mean of the Sample Project and Firm Size

By what means did the sample project arrive to your organization?	Small Size	Middle Size	All
Project was started with our customer request directly sent to us.	0,85	0,47	0,59
	5	10	15
Project was started by our organization in order to rewrite our COTS.	0,11	0,10	0,11
	1	1	2
Project was started by our organization.	-	0,78	0,78
	0	1	1
Proje was brought to our organization by third party.	0,67	0,42	0,48
	2	6	8
All	0,71	0,45	0,53
	8	18	26