E-COSMIC: A BUSINESS PROCESS MODEL BASED FUNCTIONAL SIZE ESTIMATION APPROACH

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

BY

MAHİR KAYA

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

FEBRUARY 2010

Approval of the Graduate School of Informatics

_____

Prof. Dr. Nazife BAYKAL

Director

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

_____

Assist. Prof. Dr. Tuğba TAŞKAYA TEMİZEL

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.

_____

Assoc. Prof. Dr. Onur DEMİRÖRS

Supervisor

Examining Committee Members

Assist. Prof. Dr. Altan KOÇYİĞİT (METU, II)_____

Assoc. Prof. Dr. Onur DEMİRÖRS (METU, II)_____

Assist. Prof. Dr. Aysu BETİN CAN (METU,II)_____

Dr. İdris DOĞAN (DPT)_____

Dr. Selçuk GÜCEĞLİOĞLU (BAŞBAKANLIK)_____

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: Mahir KAYA

Signature: _____

ABSTRACT

E-COSMIC: A BUSINESS PROCESS MODEL BASED FUNCTIONAL SIZE ESTIMATION APPROACH

KAYA, Mahir

M.Sc., Department of Information Systems

Supervisor: Assoc. Prof. Dr. Onur Demirörs

February 2010, 120 pages

The cost and effort estimation of projects depend on software size. A software product size is needed at as early a phase of the project as possible. Conventional Early Functional Size Estimation methods generate size at the early phase but result in subjectivity and unrepeatability due to manual calculation. On the other hand, automated Functional Size Measurement calculation approaches require constructs which are available in considerably late software development phases.

In this study we developed an approach called e-Cosmic to calculate and automate the functional size measurement based on the business processes. Functions and input and output relationship types of each function are identified in the business process model. The size of each relationship type is determined by assigning appropriate data movements based on the COSMIC Measurement Manual. Then, relationship type size is aggregated to produce the size of each function. The size of the software product is the sum of the size of these functions. Automation of this process based on business process model is performed by developing a script in the ARIS tool concept.

Three case studies were conducted to validate the proposed functional size estimation method (e-Cosmic). The size of the products in the case studies are measured manually with COSMIC FSM (Abran et al, 2007) as well as using a conventional early estimation method, called Early and Quick COSMIC FFP. We compared the results of different approaches and discussed the usability of e-Cosmic based on the findings.

Keywords: Early size estimation, Functional size measurement, Business process model.

ÖZ

E-COSMIC: İŞ SÜRECİ MODELİ TEMELLİ FONKSİYONEL BÜYÜKLÜK KESTİRİM YAKLAŞIMI

KAYA, Mahir

Yüksek Lisans, Bilişim Sistemleri Bölümü

Tez Yöneticisi: Doç. Dr.Onur Demirörs

Şubat 2010, 120 sayfa

Projelerin maliyet ve işgücü kestirimleri yazılımın büyüklüğüne bağlıdır. Bir yazılım ürününün büyüklüğüne, projenin mümkün olduğu kadar erken bir safhasında ihtiyaç duyulur. Geleneksel erken büyüklük kestirim yöntemleri büyüklüğü erken aşamada bulmakta fakat elle hesaplamadan dolayı öznelliğe ve tekrarlanamayan sonuç üretimine neden olmaktadır. Diğer taraftan otomatikleştirilmiş FBÖ hesaplama yaklaşımları yazılım geliştirme safhalarının oldukça ileri aşamasındaki yapılara ihtiyaç duyarlar.

Bu çalışmada, iş sureçleri temel alınarak fonsiyonel büyüklük ölçümünü hesaplamak ve otomatikleştirmek için e-Cosmic olarak adlandırılan bir yaklaşım geliştirilmiştir. Iş süreç

modelindeki fonksiyonlar ve her fonksiyonun girdi ve çıktı ilişki çeşitleri tanımlanmıştır. Her ilişki çeşidinin büyüklüğü, COSMIC ölçüm elkitabı temel alınarak uygun veri hareketleri atanması ile belirlenmiştir. Daha sonra, her fonksiyonun büyüklüğünü bulmak için ilişki çeşitleri büyüklüğü toplanır. Yazılım ürünün büyüklüğü ise bu fonksiyonların büyüklüğünün toplamıdır. Bu sürecin, iş süreçleri modeli temel alınarak otomatikleştirilmesi ARIS aracında bir kod sayfası geliştirilerek gerçekleştirilmiştir.

Önerilen fonksiyonel büyüklük kestirim yönteminin (e-Cosmic) geçerlemesi için üç örnek olay incelemesi yürütülmüştür. Örnek olaylardaki ürünlerin büyüklüğü hem COSMIC FBÖ hem de Early and Quick COSMIC FFP olarak adlandırılan geleneksel bir erken kestirim modeli kullanılarak elle ölçüldü. Farklı yaklaşımların sonuçları karşılaştırıldı ve bulgular temel alınarak e-Cosmic yönteminin kullanılabilirliği tartışıldı.

Anahtar Kelimeler: Erken büyüklük kestirimi, Fonksiyonel büyüklük ölçme, İş süreci modeli.

To Yasemin Çetin

ACKNOWLEDGMENTS

I wish to offer my sincere thanks and appreciation to my supervisor Assoc. Prof. Dr. Onur Demirörs for his precious help, invaluable suggestions, continuous support, guidance, criticisms, encouragements and patience throughout this study.

I would like to also express my special gratitude to Çağaçan Değer for his support and suggestions throughout my research.

I am very grateful to Özden Özcan Top, Ahmet Coşkunçay and Banu Aysolmaz for their support during the thesis study.

I would also like to thank my friends Murat Güneş, Anna Güneş, Ersin Karaman, Fatih Ömrüuzun, Rahime Belen, Yalın Baştanlar and Erdir Ungan for their friendship and encouragement.

I appreciate Ayşe Ceylan, Sibel Gülnar, Necla Işıklar and Ali Kantar in the institute for their kindness since the beginning of my M.Sc. study.

I will never forget the support of my friends Cansel Kadıoğlu, Evren Şumuer, Ali Sinan Dike and Murat Özkaptan.

I would like to express my deepest gratitude to my parents, my brother and my sisters for their love, encouragement, and support throughout my education.

Finally, special thanks to Yasemin Çetin, who endured this long process with me, for her continuous encouragement, infinite patience and moral support.

TABLE OF CONTENTS

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURE

LIST OF ABBREVIATIONS AND ACRONYMS

ARIS	: Architecture of Integrated Information Systems
BPM	: Business Process Modeling
CFP	: COSMIC Function Point
Cfsu	: COSMIC functional size unit
COSMIC	: Common Software Measurement International Consortium
COSMIC FFP	: COSMIC Full Function Point
CRUD	: Create Read Update Delete
DET	: Data Element Type
DFD	: Data Flow Diagram
E	: Entry
E&Q COSMIC FFP	: Early and Quick COSMIC Full Function Point
E&QCFFP	: Early and Quick COSMIC Full Function Point
eEPC	: Extended Event Driven Process Chain
EIAP	: Extending Independent Assessor Pool
EFPA	: Early Function Point Analysis
EFSE	: Early Functional Size Estimation
ER	: Entity Relationship
IA	: Independent Auditor
FAD	: Function Allocation Diagram
FBÖ	: Fonksiyonel Büyüklük Ölçümü
FFP	: Full Function Point
FP	: Functional Process
FPA	: Function Point Analysis

FSM	: Functional Size Measurement
FUR	: Functional User Requirement
GP	: General Process
IFPUG	: International Function Point Users Group
ILF	: Internal Logical File
LDG	: Logical Data Group
LOC	: Lines of Code
MDG	: Multiple Data Group,
MIS	: Management Information Systems
MP	: Macro Process
NESMA	: Netherland Software Metrics Association
OO	: Object Oriented
OO-FPA	: Object Oriented Function Point Analysis
OOI	: Object of Interest
PM	: Problem Management
PPP	: Perform Project Payments
PROBE	: PROxy-Based Estimating
PSP	: Personal Software Process
R	: Read
RRTT	: Rational Rose Real Time
RUP	: Rational Unified Process
Sfsu	: Scenario function size unit
SLOC	: Source Line of Code
SPR	: Software Productivity Research
TP	: Typical Process
Ufsu	: Use case function size unit
UML	: Unified Modeling Language
W	: Write
X	: Exit

CHAPTER 1

INTRODUCTION

Software industry continues to have problems related to project management. The major problems are related to difficulties of achieving high product quality and estimating effort and the time required for projects (Glass, 2003). The accuracy of size estimation is considered to be one of the most significant causes of management problems as estimated size is the basis of accurate effort prediction approaches.

It is at the beginning of the project when we most need the software product size in order to make a realistic effort prediction and a proper allocation of available time and resources. In addition to being the most reliable indicator of effort prediction models software size is frequently utilized by other quality measures to normalize and compare the process outcomes.

Among the various size measurement methods, the ones based on “functionality” are widely-used due to their earlier applicability in the software life cycle. Although “Line of Code” (LOC) based size estimation has some drawbacks, it had been widely used until Allan Albrecht introduced the idea of measuring a size of software by means of its requirements or functional specifications. Funtional size became a significant innovation since the method does not depend on the development environment and programming language of the software product (Symons, 2001).

In earlier Functional Size Measurement (FSM) methods, generation of the functional size of a software product might have been varying depending on the assumptions of the measurer. It was heavily dependent on the way how the measurer maps the software requirements represented in natural language to the elements of the FSM method. Various FSM methods and a set of related standards have been proposed and published since the original size estimation method proposed by Albrecht (1979). Modern FSM methodologies such as COSMIC (ISO/IEC 19761, 2003a), IFPUG Function Point Analysis (ISO/IEC 20926, 2003b) and Mark II Function Point Analysis (ISO/IEC 20968, 2002) can be reliably applied after the requirement specification phase.

There are also various early functional size estimation methods (Meli, 1997; Santillo and Melli, 1998; Abran, Desharnais, Oligny, St-Pierre & Symons, 2003; Conte, Iorio, Meli & Santillo, 2004; Galorath and Evans, 2006; Jones, 2007) that enable functional size prediction (in COSMIC, IFPUG or Mark II) earlier than the software requirements phase. Early functional size estimation methods entail a certain degree of subjectivity (Santillo and Meli, 1998). They are also frequently criticized of being unrepeatable and subjective (Vogelezang, 2005).

Functional size estimation starts with extracting Functional User Requirements (FURs) from software product requirements. However, researchers who worked on formalizing and automating function point measurement have been largely interested in extracting elements of FSM methods from the software artifacts such as Data Flow Diagrams (DFD), Entity Relationship (ER) Diagrams, State Diagrams and Object – Oriented Diagrams.

During the recent years, software size estimation from Object-Oriented Design artifacts has emerged as a good solution for software size estimation problems such as measurer dependency or subjectivity, low accuracy and reliability and high measurement costs (Diab, Koukane, Frappier & St-Denis, 2005). In order to overcome these problems, considerable researches have been conducted on developing software size measurement algorithms and mappings for Object-Oriented Design artifacts (Poels, 2003; Diab et al., 2005; Condori-Fernandez, Abrahao & Pastor, 2007; Levesque, Bevo & Cao, 2008). However, researchers

have not reached to an agreement on a unified definition of Object-Oriented Design artifacts to be used for automation of software size (van den Berg, Dekkers & Oudshoorn, 2005). Furthermore Object-Oriented Design artifacts needed for software size estimation might not be available in early phases of the software life cycle. Although most studies have used class diagram and sequence diagrams as input artifacts for automation of software size estimation, class and sequence diagrams may be available only considerably later during the software development process than the time when the functional size can be measured manually (Uemura, Kusumoto & Inoue, 1999). Generation of these artifacts also requires one third of all project development effort (Meli, 1997). In addition, it has been observed that most FSM automation approaches could not make their algorithm compatible with the whole measurement procedure of a FSM method (Abrahao and Emilio, 2008).

It has been hypothesized that a solution for cost effective, early size measurement was to establish an approach to measaure functional size based on process model constructs.

Business process models are frequently developed during organizational process improvement studies or system analysis phases of information system development projects. In other words, these models are available before the requirement specification phase of a software development project.

In this study to overcome the reliability and subjectivity problems, a mapping between functional size estimation components and common business process model constructs was developed and to decrease the measurement costs, the counting process was automated. The mapping and automation algorithm were also integrated into the methodology called e-Cosmic.

In e-Cosmic, the business process model elements are mapped to the main components, “functional processes” of COSMIC FSM method (Abran et al, 2007) and components of functional processes are mapped to different types of externally observable relationship types that are represented as input to and output of activities of processes. An algorithm

was defined to count the occurrences of those relationships. This algorithm was implemented in a specific process modeling environment called ARIS (Davis and Brabander, 2007).

In order to validate e-Cosmic functional size estimation method, three case studies have been conducted. The applicability of e-Cosmic functional size estimation approach was explored in the first case study. After refinements to the approach, second case study was conducted. During the second case study, a well known early functional size estimation method called Early and Quick COSMIC Full Function Point (Meli, Abran, Ho & Oligny, 2000) was also implemented and the results were compared with e-Cosmic. Third case study was conducted to observe the deviations between COSMIC FSM results and estimation results.

In Chapter 2, a literature review on related subjects is presented and discussed. COSMIC FSM method which is used as a base model is described in this chapter. In addition, early functional size estimation methods and FSM automation approaches available in the literature are summarized and compared.

In chapter 3, e-Cosmic estimation methodology is explained. Mapping between the concepts of the process model constructs and the COSMIC FSM is detailed.

In chapter 4, we described the case studies that we have performed to validate e-Cosmic Functional size estimation method.

In chapter 5, significance and contributions of the study are presented and suggestions for future work are provided.

CHAPTER 2

RELATED RESEARCH

This chapter presents the results of the literature review on software size measurement/estimation methods, early software size estimation approaches and automation approaches of software functional size measurement.

2.1. Software Size Measurement/Estimation Methods

Since the early computer era of 1940s, software size measurement has been a prominent but difficult activity. Software size is necessary to make a reliable project plan and properly allocate time and resources. In addition, size may help post-assessment about development effort by providing feedback on quality and productivity, on which various software measures depend.

There has been considerable research in the literature on the accurate and complete estimation of software size. These various approaches can be classified as technology-independent methods such as Function Point Analysis (FPA) method and technology-dependent methods such as LOC or number of classes.

Although counting program code is the oldest method and is used widely, the debate on how good it represents the size of software has come into prominence. In addition, there is not a consensus on definition of the measurement procedure such as counting comment line or only executable statements. The advantages and disadvantages of the LOC are shown in Table 1 (Galorath and Evans, 2006).

Table 1: Advantage and Disadvantage of SLOC

Advantage	Disadvantage
Easy to compute	It heavily depends on how code is written
Intuitive	Multiple definitions for what constitutes a line of code.
Granular	Varies dramatically based on technology used
Natural by-product of process	It becomes available at the end of the development.
	Some languages difficult to count automatically

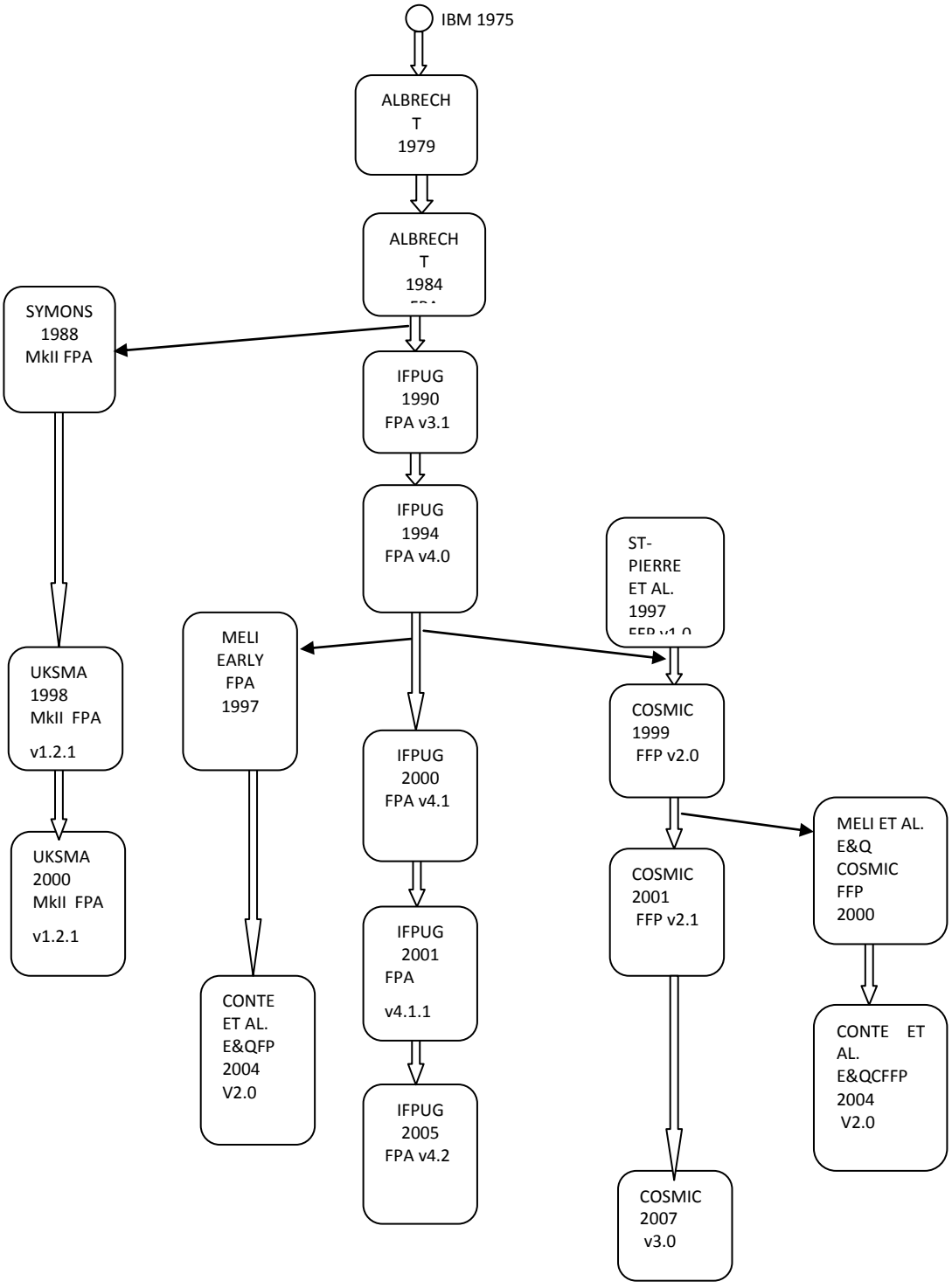
Source: Based on Galorath and Evans (2006), edited by the author.

In the late 1970s, in order to overcome deficiencies of source code based approaches, Albrecht (1979) proposed a method based on measuring amount of functionality delivered to user. The size is measured from a functional, or user, point of view. The Function Point technique is independent of the computer language, development methodology, technology or the capability of the program developers. Symons (2001) stated that "At the time it was a genuine breakthrough in thinking by providing the first method for sizing software which was independent of the technology to be used for its development" (Symons, 2001: p413).

Albrecht's model counts software product in terms of four characteristics; namely External Inputs and Outputs, User Interactions, External Interfaces and Files used by the system. After identification of these types of components in a specification, they are weighted from 3 (simple) to 15 (complex). The Function Point Method has evolved and got refined in course of time. The result of these refinements is the "counting manual" produced by IBM in the early eighties. International Function Point Users Group (IFPUG) was formed in 1986. This organization produced its own Counting Practices Manual to clarify and standardize rules for the application of FPA. Since the first publication of IFPUG FPA, each release of these IFPUG publications has provided refinements to the technique originally presented by Albrecht. The IFPUG (Release 4.2) was published in 2005.

There are four functional size measurement methods that are certified by ISO/IEC. These are IFPUG FPA, Mark II FPA, Netherland Software Metrics Association (NESMA) FPA and

COSMIC FFP. NESMA and IFPUG methods are alike and Mark II method is used less frequently (van den Berg et al, 2005). The history of FPA methods are shown in Figure 1 (Levesque et al, 2008). The detailed discussion of differences between FPA methods, and their evolutions are out of the scope of this thesis. COSMIC, details of which is given later in this section, is selected as FPA method in this thesis study because of the fact that it is easy to apply, suitable for MIS applications, has a measurement procedure more appropriate for counting business process model and the definition of entity attributes is not required for counting.



Source: Based on Levesque et al. (2008), edited by the author

Figure 1: Evolution of the FSM methods

The COSMIC FSM method is accepted as a second generation size measurement method. The word “COSMIC” stands for “Common Software Measurement International Consortium”. The COSMIC Function Point method evolved from “Full Function Point” method coupled with some aspects of Mark II function point (Jones, 2008). A group of software metrics experts established COSMIC in 1999 (Abran, 1999). It has been improved and new versions have been released since then.

The COSMIC method was designed as a standardized method of measuring a functional size of software from the functional domains that are business application (or MIS) software and real-time software (Abran et al., 2007). The functional size of any component of software can be measured as seen by direct users of that component. On the other hand, it has not been suitable for measuring complex mathematical algorithms such as expert systems and simulation systems. In addition to this, it does not take into account neither the influence of complexity arising from technology nor the influence of the number of data attributes per data movement on software functional size.

The measurement process of the method includes three phases: the measurement strategy, the mapping of the concepts and the measurement phase that represents value assignments. In the first phase, the purpose and the scope of the measurement must be identified. Then, functional users, which send or receive data to or from the functional process, must be identified. FURs must be extracted from software artifacts. Finally, the “levels of granularity” should be identified in the beginning of the measurement. Scope and purpose affect software size in terms of both “user viewpoint” or “developer viewpoint” and entry-exit pair across the boundary. These terms also help determining the boundary of the application.

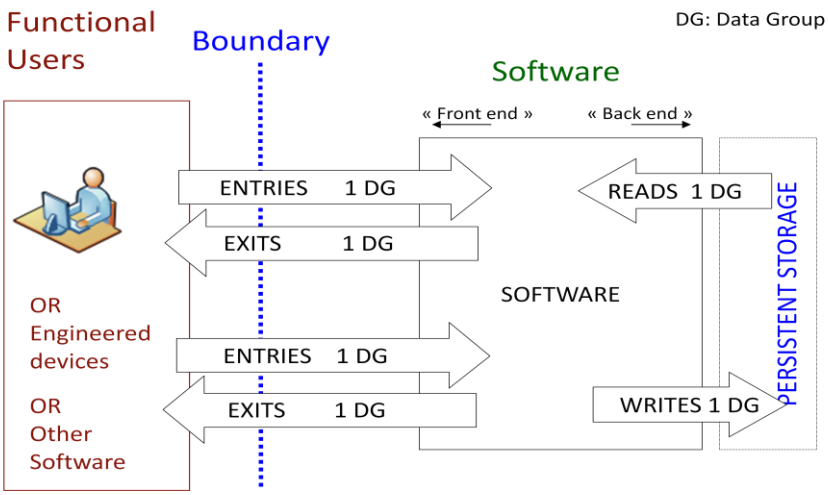
In the mapping phase, functional processes, which are elementary components of a set of FURs, must be identified. It comprises a unique, cohesive and independently executable set of data movements. Functional user triggers the functional process by an entry data movement. Then, the data group and the object of interest (OOI) should be determined. A data group is a set of data attributes that are distinct, non empty, unordered and non

redundant. OOIs can be any “entity” which is related with FURs. It is optional to define attributes of a data group (Abran et al., 2007).

In the measurement phase, the data movements for each functional process should be identified. Data movements are the Entries, eXits, Reads and Writes crossing the boundary between the functional user and the application (ISO/IEC, 2003a). A visual representation can be seen in Figure 2. After identification of all data movements, we can apply measurement function as assigning 1 CFP to each data movement of the functional process, when all functional processes are aggregated, the functional size of the measured piece of the software is generated.

The data movements are described as follows:

- An Entry (E) is a movement of a data group from the user’s side of the boundary to the inside of the boundary
- An Exit (X) is a movement of a data group from inside the boundary to the user side.
- A Read (R) is a movement of a data group from the persistent storage to the functional process which requires it.
- A Write (W) is a movement of a data group from the functional process to the persistent storage.



Source: Based on Abran et al. (2007), edited by the author

Figure 2: Data movement types and their relationship with software

2.2. Early Software Size Estimation

The most important advantage of FSM methods is their usability when FURs of business application are defined. However, project managers also need a method for planning and managing the tasks of the requirements phase as requirement specification tasks comprises from 15 % to 40 % of the total work effort (Meli, 1997).

In order to overcome this problem a few techniques for early functional size estimation are developed. In this section we will describe each early functional size estimation method to represent their applicability. Strengths and weaknesses of them will be identified.

2.2.1. Jones Very Early Size Predictor

This is an estimation method developed by Capers Jones. Jones (2007) tried to transform function point metric from sizing a very difficult task into one that is now not only easy to apply but comparatively accurate as well.

Most software cost estimating tools need to identify the project in terms of scope, class and type. The method utilizes taxonomy for defining software projects. Early size approximation of function points are produced by utilizing this project identification mechanism. Although this basic method of sizing is far from producing reliable results for serious cost estimating purposes, if we do not have enough information about the project it emerges as a good solution to find project size. The examples of the scope, class and type of taxonomy for guessing at the approximate size of the software are depicted in Table 2.

In order to apply this superficial sizing method, we firstly identify numeric list values of the project in terms of scope, class, and type factors and then apply the following formula proposed by Jones (2007):

$$\text{Size} = (\text{Scope} + \text{Class} + \text{Type})^{2.35}$$

Table 2: Examples of Scope, Class, and Type Values

Scope:	Class:	Type:
1) Subroutine	1) Individual software	1) Nonprocedural
2) Module	2) Shareware	2) Web applet
3) Reusable module	3) Academic software	3) Batch (not database)
4) Disposable prototype	4) Single location – internal	4) Interactive
5) Evolutionary prototype	5) Multi location – internal	5) Interactive GUI or web -based
6) Standalone program	6) Contract project – civilian	6) Batch database
7) Component of a system	7) Time sharing	7) Interactive database
8) Release of system	8) Military services	8) Client/server
9) New system	9) Internet	9) Mathematical
10) Compound system	10) Leased software	10) Systems
	11) Bundled software	11) Communications
	12) Marketed commercially	12) Process control
	13) Outsource contract	13) Trusted system
	14) Government contract	14) Embedded
	15) Military contract	15) Image processing
		16) Multimedia
		17) Robotics
		18) Artificial intelligence
		19) Neural net
		20) Hybrid: mixed

Source: Based on Jones (2007)

This rough approximation method gives a limited size value. If we use the highest numeric list values of scope, class, and type, the maximum size of a project may be approximately 7675 FP. According to this value, we would conclude that the method is not suitable for large software projects. However this approximation method is enough to see whether an application will be a trivial one. In addition, Jones (2007) suggested that this taxonomy would be suitable for other metrics such as object points, CFPs, or engineering function points by using the proper power for formula. Since this empirical formula utilizes only a fundamental taxonomy to try to find applications with similar patterns, it can be applied on thousands of projects in existence to come up with appropriate power for specific projects.

2.2.2. Standard Component Sizing

Standard component sizing depends on identification of a number of key components and then comparison to the collected past data related to this key components (Vogelezang, 2005). An estimation of average and extreme size for key components can be produced by comparison of historical data on similar components, so data of previous projects is required to apply the method. Another method suggested by Putnam and Myers relies on Fuzzy logic concepts.

Fuzzy logic is used as a systematic way to perform comparison with past work. It was used by Putnam and Myers for software sizing purposes after adaptation to this field. According to the project size, systems built previously are divided into six sub-categories. In order to represent quartiles, four ranges are created within each size category. A new information system is compared to the past projects, thus depending on the expert opinion the size of the new system can be estimated (Vogelezang, 2005).

2.2.3. Proxy Based Estimating

PROxy-Based Estimating (PROBE) method has evolved from the similarity based approach. Referring to Humphrey (1995), Vogelezang (2005) states that if a component needed to be developed is similar to the one built previously, the result of its software metrics will resemble to the previously completed work.

Historical data is needed to apply the method. A size range is assigned to each conceptual function. In order to estimate size and effort in the Personal Software Process (PSP) which is a discipline that enables software engineers to monitor, test, and improve their own work, PROBE method can be used.

In the PROBE method, individual engineers break down past works into individual components. A type such as data or logic is identified for each component and then from very small to very large a size is counted and assigned to each component. Before a new project is started and determined a resource allocation, the size estimation of the project must be finished. The project is firstly separated into tasks according to the types and sizes and then a formula is applied in order to estimate size.

2.2.4. Wideband Delphi

Software size estimate mostly needs judgment of the estimator. Lacking rigor, this process may sometimes generate biased results about size of the software product (Galorath and Evans, 2006). In order to eliminate these disadvantages, one can apply Delphi estimation method. That is based upon software size estimations of expert teams and individuals. These groups and individuals start with the same description of the task and produce estimates anonymously. The average size is calculated based on estimation of the experts by a moderator and this average and all other estimates are given to the experts in order to repeat estimation. This process is repeated until the estimates get close enough and a consensus is reached (Boehm, 1981). In addition, this method might take much time to agree on a size; on the other hand, experts may not reach a consensus.

2.2.5. Statistical Sizing

The method relies on asking questions to the developer in order to determine a range of potential sizes that is classified by “least”, “likely”, and “most” (Galorath and Evans, 2006). In this method, estimator prepares a questionnaire to handle all expected and unanticipated situations during the development phase of the software product.

The questions are asked to the developer consecutively since the order of the question is important. The initial question most likely would be that “what do you think the size of the software product will be”. Then, other questions are designed according to capturing all of the risk related to the product. The classification of the results is depicted as follows:

- Least =” What is the best case size?”
- Likely =”What is the expected size?”
- Most =”What is the worst case size?”

Galorath and Evans (2006) determined a formula in order to apply statistical sizing method as follows:

Expected value = [Least + (4*Likely) + Most]/6

Standard deviation = Most – Least

Expected size for any probability level can be calculated by using the standard deviation and “bell curve” probability table. This method is also known as Three-point estimation technique (Meli and Santillo, 1999).

2.2.6. FPA_i

This method is structured from many approaches so as to estimate the functional size of the software product (Vogelezang, 2005). Not only does this method benefit from some feature of the fuzzy logic and standard-component sizing but it is the adjusted convergence midpoint of the Delphi method and Early and Quick FPA method. Weight factors of the requirements are assigned by estimators in this approach. Then, a minimum, an average, and a maximum estimate values are determined for each weight factor. When we compare this method to the Early and Quick FPA and Delphi method, this method is less complex than Early and Quick method and it needs less time than Delphi method.

2.2.7. Early Function Point Analysis

Early Function Point Analysis (EFPA) method was introduced in 1997 (Meli, 1997), and then it is refined and exemplified by Santillo and Meli (1998). Early Function Point technique is not perceived as alternative measurement method to the FPA method since it is developed

to produce a fast and early estimate as compared to the FPA (Meli, 1997). The measurement procedure of the method is based on using both analogical and analytical classification of functionalities. As considering this approach, we can see that the method produces better software product estimation compared to previously covered approaches.

The method allows estimators to identify software objects, which are composed of logical data and functionalities aspects, at different levels of detail. This aspect makes the method more robust because it is easy to increase detailed level of the software application brunch that is previously produced; on the other hand, little or nothing may be known about another software branch that is completely innovative (Santillo and Meli, 1998). The method enables estimators to use all information about the software application by supporting the different detail level of the software objects. There is a minimum, an average, and a maximum estimate for each type of object. Early & Quick IFPUG Function Point (E&QFP 2.0), which is an evolution of this technique, was released in 2004 (Conte et al., 2004). The technique is performed through statistically and analytically validated tables of values. There are 6 software objects in E&QFP 2.0. From high level to low level, these objects are macro process, general process, typical process, functional process, multiple data group and logical data group. Macro process is a set of two or more average general processes. A general process contains a set of two or more average functional processes. The methodology defines a functional process as the elementary processes of the standard FPA. A typical process is a particular case of general process composed of "*create*, *read*, *update*, *delete"* (CRUD) and *list* (L) operations. Finally, logical data groups are presented to be a group of logical data attributes. Multiple data groups consist of a set of two or more logical data groups.

A set of size values based on statistical/analytical tables is determined for each software object, and then these values are summed in order to find the overall result (Conte et al., 2004). These methods rely on the estimator's ability to identify and assess the software objects accurately and completely. Therefore, this situation increases the subjectivity of the method.

2.2.8. Early & Quick COSMIC-FFP

In order to extend software domain of the EFPA method, EFPA research group has designed E&Q COSMIC FFP based on present COSMIC FFP (Abran et al., 2003) principles. Such an extension was needed for EFPA was based on IFPUG FPA, which is suitable mostly for MIS domain and thus requires a domain expansion.

Meli et al. (2000) stated that Early & Quick estimation must not be perceived as an accurate measure of the software functional size, but only an estimation of that measure allowed to be produced for a wide range of software at early stages of the development life cycle. E&Q COSMIC FFP 2.0 is released as a new proposal of the first version (Conte et al., 2004).

In order to apply the measurement procedure of the COSMIC FFP, a new software product has to be represented according to the software breakdown structure (Meli et al., 2000). The software product is decomposed towards finding a single node consisting of unique and ordered set of data movements in the function tree and then accepting those leaves as an input for measurement process to produce final size for the software item under measurement. On the other hand, in the early stages, it is not possible to distinguish the single data movements because of the fact that detailed level of information is not available. Thus, forecasts of average process size, at the intermediate and top levels, are assigned. Then, final result is produced by aggregating the intermediate results. Moreover, the size of functional process in the COSMIC FFP is not bounded as E&Q COSMIC FFP. The component ranges and the numerical assignment for the E&Q COSMIC FFP 2.0 Release Candidate is represented in Table 3. But it should be noted that this technique also entails a certain degree of subjectivity as EFPA does.

Table 3: E&QCFFP 2.0 elements ranges and numerical values

Type	Level	Ranges / COSMIC Equivalent	Min. CFSU	Most likely CFSU	Max. CFSU
Functional Process	Small	1-5 Data movements	2.0	3.9	5.0
	Medium	5-8 Data movements	5.0	6.9	8.0
	Large	8-14 Data movements	8.0 10.5 14.0	8.0 10.5 14.0	8.0 10.5 14.0
	Very Large	14+ Data movements	14.0	23.7	30.0
Typical Process	Small	CRUD (Small/Medium processes); CRUD + List (Small processes)	15.6	20.4	27.6
	Medium	CRUD (Medium/Large processes); CRUD + List (Medium processes); CRUD + List + Report (Small processes)	27.6	32.3	42.0
	Large	CRUD (Large processes); CRUD + List (Medium/Large processes); CRUD + List + Report (Medium proc's)	42.0	48.5	63.0
General Process	Small	6–10 generic FP's	20.0	60.0	110.0
	Medium	10–15 generic FP's	40.0	95.0	160.0
	Large	15-20 generic FP's	60.0	130.0	220.0
Macro Process	Small	2–4 generic GP's	120.0	285.0	520.0
	Medium	4–6 generic GP's	240.0	475.0	780.0
	Large	6–10 generic GP's	360.0	760.0	1,300.0

Source: Based on Conte et al. (2004)

2.2.9. Approximate COSMIC- FFP

This technique is based on using an average value produced from previous software applications for the size of a functional process. An approximate version of the COSMIC-FFP method was designed in order to generate a software size estimation at an early phase of the software development process, when information needed to produce a size estimate by means of the detailed rules in the Measurement Manual is not available (Abran et al., 2003). The main aim of this method is to create a concept at a higher level of abstraction than the Data Movement, and then functional process is defined as first higher level abstraction. The size estimation of an application is produced by multiplying the quantity of the functional processes by the average size of a functional process. Subsequent COSMIC (2007) measurement manual excluded this section from the context. However, conducted literature review has failed to identify the reason for this exclusion.

2.3. Software Size Automation Approaches

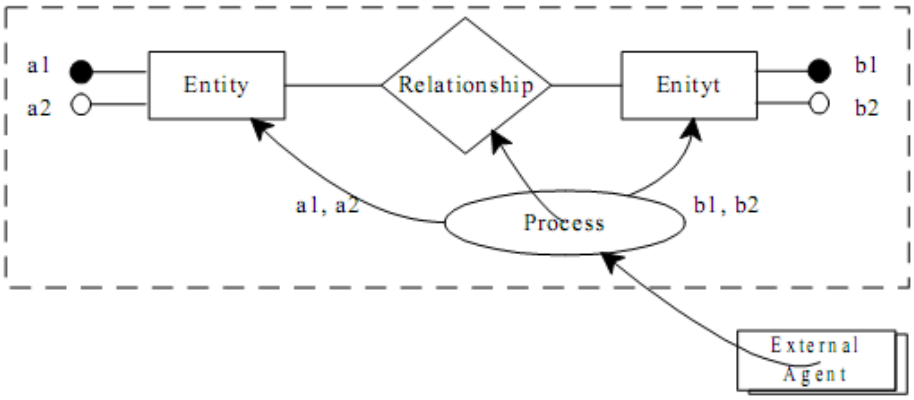
In this section, we present the automation approach on measurement procedure of the FSM methods. During the literature survey, we identified a number of attempts to automate measurement procedure of the FSM methods by means of the software models. Ever since the first FPA method was introduced, it has evolved continuously and new methods were proposed. All of these methods define software size estimation procedure in a descriptive way that lacks a formal foundation (Fetcke, 1999). Many researchers have tried to translate the informal counting rules expressed in natural language that lacks formalization (Lamma, Mello & Riguzzi, 2004). Fenton and Pfleeger (1997) defined the importance of the rules in terms of support for consistency in measurement and provision of a basis for interpreting data.

It is important to make automation of the function point measurement possible so as to produce consistent measurement results with minimum effort (Uemura et al., 1999). Automation of the measurement procedure enables support measurement repeatability, reductions in measurement variance, removes estimator's dependency and estimator's need for advanced measurement training and eliminates measurement related costs.

Marin, Giachetti & Pastor (2008) conducted a comprehensive survey on measurement process approaches that use conceptual models. This survey indicated eleven proposals of functional size measurement procedures based on COSMIC FSM. This survey has been one of the most important and detailed survey on measurement process proposals so far. In addition to these, we have examined other proposals such as Fectke et al. (1997), Lamma et al. (2004), Rask (1991), Poels (2003), Uemura et al. (1999), van den Berg et al. (2005), Hericko et al. (2006) and Diab et al. (2005). Hericko et al. (2006) also criticized the proposals of Fetcke et al. (1997), Uemura et al. (1999). We selected nine proposals for detailed explanation in order to represent differences of the models.

Proposal of Lamma et al. (2004)

A knowledge based approach for the automated measurement of the Function Point metrics is proposed by integrating ER diagram and DFD. FPA method rules were specialized according to the ER-DFD model. Proposed model is implemented in Prolog that automatically counts Function Point by analyzing the graph. The example graph is depicted in Figure 3.



Source: Based on Lamma et al. (1997)

Figure 3: Example of an ER-DFD diagram.

Formal rules for counting Function Points on ER-DFD are identified. Every process in the model is defined as an elementary process of the FPA method, every attribute of an entity is accepted as Data Element Type (DET), and the direction of arrow between entity and process of DFD represents the transaction type such as EI, EO, or EQ. This model has not

been accepted widely because of the fact that conceptual model representation of the software products has already transformed to the object oriented area. In addition to this, Rask (1991) proposed algorithms to generate function points from DFD.

Proposal of Bevo et al. (1999) and Jenner (2001)

The characteristics of these proposals are very similar. Both of them present a mapping between COSMIC FFP concepts and Unified Modeling Language (UML) based specifications. Bevo et al. (1999) used diagrams of use cases, scenarios, and classes as input artifacts but proposal of Jenner (2001) differs from previous work in terms of using sequence diagrams instead of scenarios which are sequence of interactions that occur inside a use case. This difference shows its effect in granularity level of use cases. Bevo et al. (1999) defined a mapping between use cases and COSMIC FFP functional processes. The boundary of the system is represented as a use case diagram. While scenarios represent data movements, a data group is represented as a class. The attributes of those classes constitute attributes of a data group. Each actor of the UML notations represents functional user. On the other hand, Jenner (2001) argued that a mapping of use cases on to functional processes may include some shortcomings. Jenner (2001) provided appropriateness of accepting a sequence diagram for each use case and each use case as a sequence of functional process, states Marin (2008). Poels (2002) introduced this conflict that is raised between Bevo et al. (1999) and Jenner (2001) as well.

Proposal of Diab et al. (2005)

Diab et al. designed a software tool named μcROSE in order to automate functional size estimation of real time applications for Rational Rose Real Time (RRRT) models. The ROOM charts used for input are a type of state chart diagram. The model consists of a set of formal rules. The RRRT model has two different viewpoints such as structure and behavior. The structure of an RRRT is composed of capsules, protocols, and data classes. A capsule is an active entity. A protocol represents a set of possible messages exchanged between capsules. And the dynamic aspects of each capsules is specified with extended finite state machine. Table 4 depicts the correspondence between COSMIC-FFP and RRRT concepts.

Table 4: Mapping between COSMIC-FFP and RRRT concepts

COSMIC-FFP concepts	RRRT concepts
Layer	Set of capsules
Boundary	Conceptual boundary of a set of capsules
Data Group	Message or a data type
Functional Process	One or several transitions
Triggering event	Arrival of an external message
Data movements	Actions and messages

Source: Based on Diab et al. (2005)

It is the first tool to provide automatic measurement of COSMIC FFP. The validation phase of the µcROSE with an expert emerged as an important behavior of the COSMIC-FFP. One of them is that µcROSE identifies one data group for all simple attributes of a capsule; however, the expert identifies one data group per simple attribute of a capsule. We want to indicate that the number of data groups determines the number of data movements (Entry, eXit, Read, Write).

Proposal of Azzouz and Abran (2004)

The proposal resembles previous works such as Bevo (1999) and Jenner (2001). The main difference of the proposal is to make early indicators of functional size measurement possible. The proposal defines three levels of the size estimation concept. The bottom level describes the detailed functional size measurement as in Table 5. Furthermore, because of the fact that there is no UML equivalence of the layer, this problem has been overcome by requesting information from the user manually. The developed tool used artifacts of the application that is produced with the Rational Unified Process (RUP). The solution mapping of the proposal different from previous works is that a new UML stereotype is defined in order to distinguish a trigger event from a simple message in use-case diagrams (Azzouz and Abran, 2004).

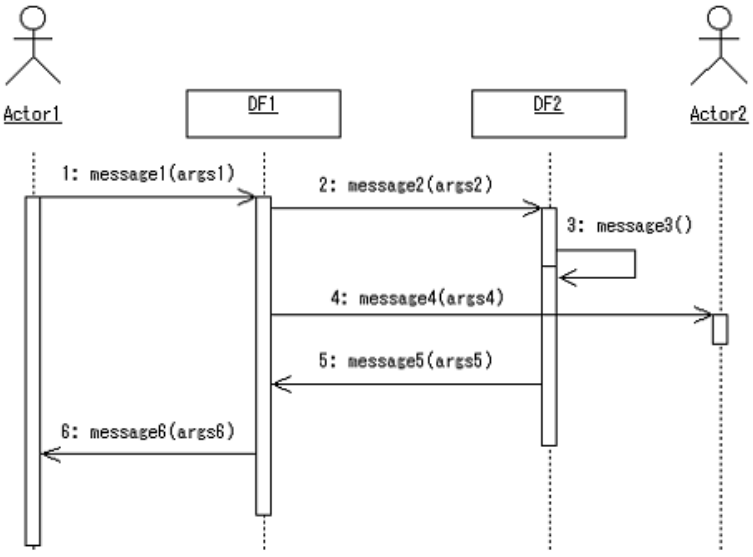
Table 5: Early and detailed size indicator

Development phase	RUP artifacts used	Unit Convention
Business modeling **Requirement analysis**	Use-case diagram	Use case function size unit (Ufsu)
Analysis / Design	Scenarios	Scenario function size unit (Sfsu)
Analysis / Design	Detailed scenarios	Cosmic function size unit (Cfsu)

Source: Based on Azzouz (2004)

Proposal of Uemura et al. (1999)

Uemura et al. (1999) proposed detailed FPA measurement rules for the design specifications which are based on the UML. The proposal is also supported with a tool. It focuses on OO-FPA mapping like previous attempts such as Fetcke et al. (1997). Class diagrams and sequence diagrams represent the input of the measurement process. Object of the sequence diagrams defines boundary of the measurement. The objects with methods that change the state of other objects are mapped to ILF. Others are considered as EIF. There are five different sequence diagram patterns restricted by some common assumptions in order to identify type of the transactional functions. The example of the pattern 5 is depicted in Figure 4 (Uemura et al., 1999).



Source: Based on Uemura et al. (1999)

Figure 4: Example of the sequence diagrams pattern

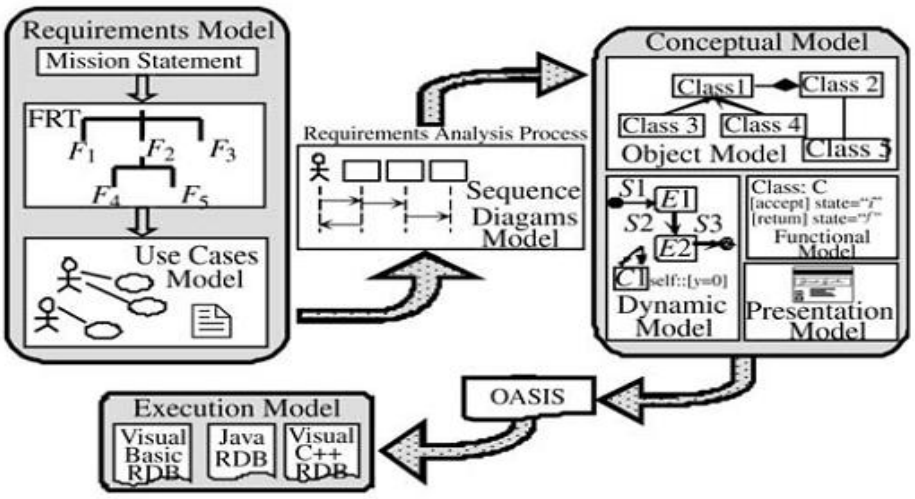
Although the five patterns is considered enough for determining the transactional function without relying on any additional information, there are some drawbacks. The messages without any argument are not counted and the case of a message being used in different sequence diagrams is not explored in depth (Hericko et al., 2006).

Proposal of Poels (2002)

The proposal is to provide a set of rules for mapping COSMIC FFP concepts to the Object Oriented (OO) methodology such as MERODE (Snoeck et al., 1999) composed of business model and services model. Poels (2002) supports that event-based OO enterprise modeling that uses MERODE is suitable for real world and system events. The business model includes class diagrams, an object-event table that describes which type of business events affect which type of business domain objects, and state transition diagrams model. The information system service model provides end-user event transactions to the enterprise model (Poels, 2003). The proposal defines rules of the layered architecture. We can describe functional user of the business layer as objects of the services layer; on the other hand, the functional user of the service layer corresponds to objects of the user interface layer. Functional process of the business model is a set of class methods which are invoked by the occurrence of a type of business event. Functional process of the service model is a non-persistent service object. Data movements are identified according to type of the different class methods (Poels, 2003).

Proposal of Condori-Fernandez et al. (2007)

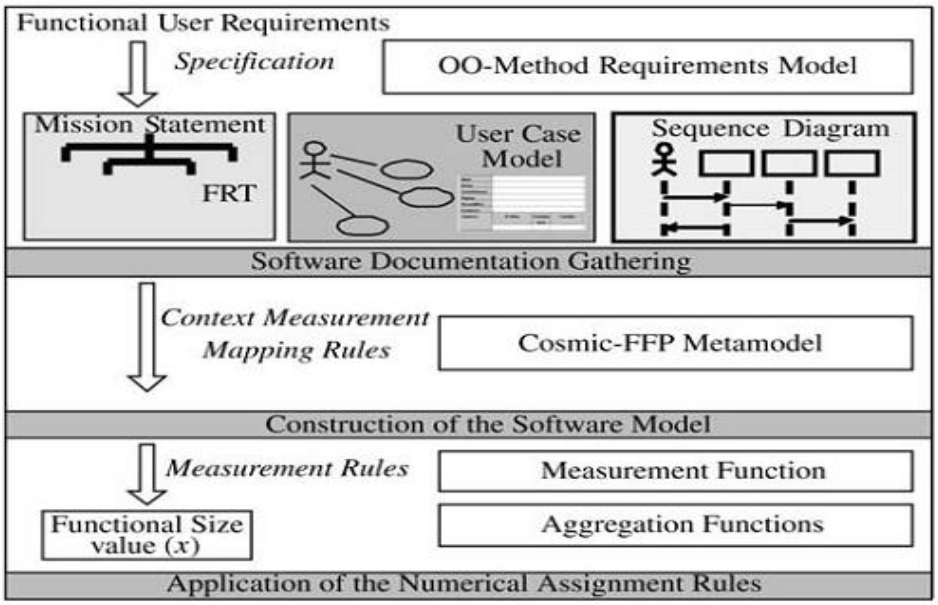
Condori-Fernandez et al. (2007) provide an FSM procedure, called RmFFP, which is defined in a systematic way in order to estimate the functional size of a software product from its corresponding requirements specification in the OO-Method approach. As shown in Figure 5, the OO-Method generates software application in an automatic way on different platforms by means of model transformations. OO-Method is comprised of the Requirement Model, which represents the system's functionality, the Conceptual Model, which represents static and dynamic properties of the functional requirements, and the Execution Model, which provide transition from problem space to solution space.



Source: Based on Condori-Fernandez et al. (2007)

Figure 5: The software production models in OO-method

There are three important steps to needed for the application of the measurement procedure. These are software documentation gathering, construction of the software model, and application of the numerical assignment rules. The measurement procedure of the proposals is depicted in Figure 6 (Condori-Fernandez et al. 2007). Use case diagram boundary with actors is accepted as system boundary to be measured. Each primary and secondary use cases that must be leaf node of the function refinement tree are accepted as functional process. Each object of the sequence diagram that includes actor, control class, or entity class is accepted as a data group. The messages of the sequence diagrams are accepted as data movements.



Source: Based on Condori-Fernandez et al. (2007)

Figure 6: Measurement procedure application

Proposal of Levesque et al. (2008)

The proposal is based on UML specifications such as use case diagrams and sequence diagrams in order to generate functional size of the software product. However, the difference of the proposal from previous work is that Levesque et al. (2008) also considers data manipulation type of the functional process type in order to take into account the complexity of the process.

Comparison of proposals of the functional size automation is given in the Table 6. The table provides an overview of the reviewed methods in terms of the measurement method they are based on, the input artifacts that they make use of and whether the process has been automated by creating and implementing a script. Table 6 also covers whether the discussed methods have been validated in the relevant studies.

Table 6: Comparison of proposals of the functional size automation

Proposal	FSM method	Input Artifacts	Automated	Validated
Lamma et al. (1997)	IFPUG FPA v 4.0	ER-DFD Diagrams	FUN (written in Prolog)	No
Bevo et al. (1999)	COSMIC FFP v. 2.0	Use Case, Scenarios, and Class Diagrams	Metric Xpert	Yes (case study)
Jenner (2001)	COSMIC FFP v. 2.0	Use Case, Sequences and Class Diagrams	No	Yes (case study)
Diab et al. (2001)	COSMIC FFP v. 2.2	RRRT model artifacts	µcROSE	Yes (case study, COSMIC experts)
Azzouz and Abran (2004)	COSMIC FFP v. 2.2	RUP artifacts	COSMIC-RUP	Yes (case study)
Uemura et al. (1999)	IFPUG FPA v 4.0	Classes, and Sequence Diagrams	Yes	Yes (Case Study)
Poels (2002)	COSMIC FFP v. 2.1	MERODE model artifacts	No	Yes
Condori-Fernandez	COSMIC FFP v. 2.2	OO-Method requirements model artifacts	RETO (not ready)	Yes (Case Study)
Levesque et al. (2008)	COSMIC FFP v. 2.1	Use Cases and Sequence Diagrams	No	Yes (Case study)

During the literature survey we observed two important improvement opportunities related to early functional size estimation. First, lack of measurement manual of early functional size estimation methods makes the reliability of them directly proportional to the estimator's ability and thus introduces subjectivity to the measurement process. An objective formulation needs to be placed at the core of a proposition for a new functional

size estimation method, so that formal representation of estimation methods is possible. Such formalization requires a passage from the business realm to the software realm. A potential candidate for such transition is business process model. FSM from the business process models is a new issue and no approach and tool can be found in the literature that automatically generates functional size of the software product from business process models.

Secondly, it has been observed that there is considerable room for improvement by automation of the measurement process. On the application side, such formalization is possible through the automation of the size estimation process. Automating the FSM is a challenging but a rewarding research area as well. Its challenge is mostly due to the difficulties related to the identification and representation of method elements accurately. The rewards include minimizing effort and time spent for measurement, eliminating errors that depend on measurer's ability and increasing repeatability.

In order to perform these improvements, we develop the early functional size estimation method based on measurement manual of COSMIC FSM method in the next chapter.

CHAPTER 3

E-COSMIC: FUNCTIONAL SIZE ESTIMATION APPROACH

Components of the design and implementation of the e-Cosmic method are grouped under three headings and they will be explained in detail respectively.

1. **Business Process Model:** Business Process Model (BPM) description is used in model extraction that is input to the e-Cosmic FSE method. BPM elements and diagrams were investigated to determine COSMIC FSM method elements and transform them to e-Cosmic FSE method. The automation requirements of the methodology have also been taken into account during investigation.

2. **Method Specification:** A method for generating early functional size estimation on business process model using eEPC diagram, function allocation diagram and its elements was developed and formalized based on the COSMIC FSM method principles.

3. **Method Implementation:** A script that makes use of the specified e-Cosmic FSE method and documents was written in this study. The script calculates functional size based on the e-Cosmic formalism and reports the results.

3.1. Business Process Models

EFPA leads to a certain degree of subjectivity due to the fact that “The reliability of the EFPA is directly proportional to the estimator’s ability to recognize the components of the

system as part of one of the proposed classes" (Santillo and Meli, 1998: p7). Hence, early size estimation method developers suggested that the expression of user requirements should be formalized as much as possible. Demirörs and Gencel (2004) claim that business process models may help this formalization. In order to decrease the subjectivity of early size estimation methods, definition of requirements has to be formalized. Since business process models enable the identification of software objects at higher abstraction levels, they are very useful to create a structured estimation process and a standard guideline.

In e-Cosmic, BPM was utilized as an approach for early software functional size estimation. More specifically, eEPC Modeling was used with Architecture of Integrated Information Systems (ARIS) concept for this study, so that roles, inputs, outputs and their relationships could be shown precisely, and functions could be presented with their triggering event.

Essentially there are four types of objects used in the eEPC. These are events, functions, rules and resources (data, entity, organization, and system). An event can trigger a function or a function can produce an event, so combinations of events and functions in a sequence produce eEPCs. Triggering multiple events or functions entails logical operators which are already part of the modeling notation.

Alternative or parallel paths are modeled with logical operators, for example, the AND logical operator is used when all paths are parallel and functions are done simultaneously, the XOR is used to choose only one of alternative paths, the OR logical operator is used when one or more alternative paths can be followed simultaneously. These are some basic examples; however, more complex expressions can be formed according to needs. eEPCs can be hierarchically structured across any number of levels by assigning more detailed eEPCs to every function within an eEPC; thus, it is easy to show sub-processes (Davis and Brabander, 2007).

In addition to business process models, two important concepts are used for the formulation of the proposed e-Cosmic FSE Method. These are Function Tree and Function Allocation Model. We need to define process purpose of the measurement model before

the definition of function tree. The process purpose is a high level description of the nature and purpose of the system. The goal of this phase is to define what the system will and will not do.

The function tree, on the other hand, depicts the hierarchical decomposition of the business functions as a module representation. It is an organization of external functions and it does not say anything about the internal process of functions. In addition to this, it consists of main business process flow and its leaf nodes are function allocation diagrams that represent the function of desired model with inputs, outputs, and relationships.

Function Allocation Model is used for modeling the functional requirements of system from the view of the user. It represents relationships between function and its environment (Davis and Brabander, 2007). This model is composed of organization, data, systems, knowledge, products and general resources subcomponents. However, this framework covers business processes at a high level. Therefore, we specialize these subcomponents so that they can be related to software size estimation approach. We use the concept of entity to cover data aspects of business process model. The concept of roles refers to organization. Application system is used to account for existing systems of the business process model whereas documents are used to cover general resources. These resource objects, namely entity, roles, application system and document, are allocated to functions in the Function Allocation Diagram (FAD) in the eEPC.

We design FAD based on "CRUDL" to describe whether a process creates, reads, updates, lists or deletes a data object. Object connection to the function is based on these simple relationships depicted in Table 7. FAD is summarized in Figure 7. Functions and some of its relationships are depicted.

Table 7: Data Object Relationship Types

Connection Direction	Relationship	Base operation
Input connection (from data object to function)	Is input for	Reads

Table 7 (continued)

	Reads	Reads
	Uses	Updates, Lists
	Views	Lists
Output connection (from function to data object)	Has output of	Creates
	Changes	Updates,Lists
	Deletes	Deletes
	Views	Lists
	Archives	Creates

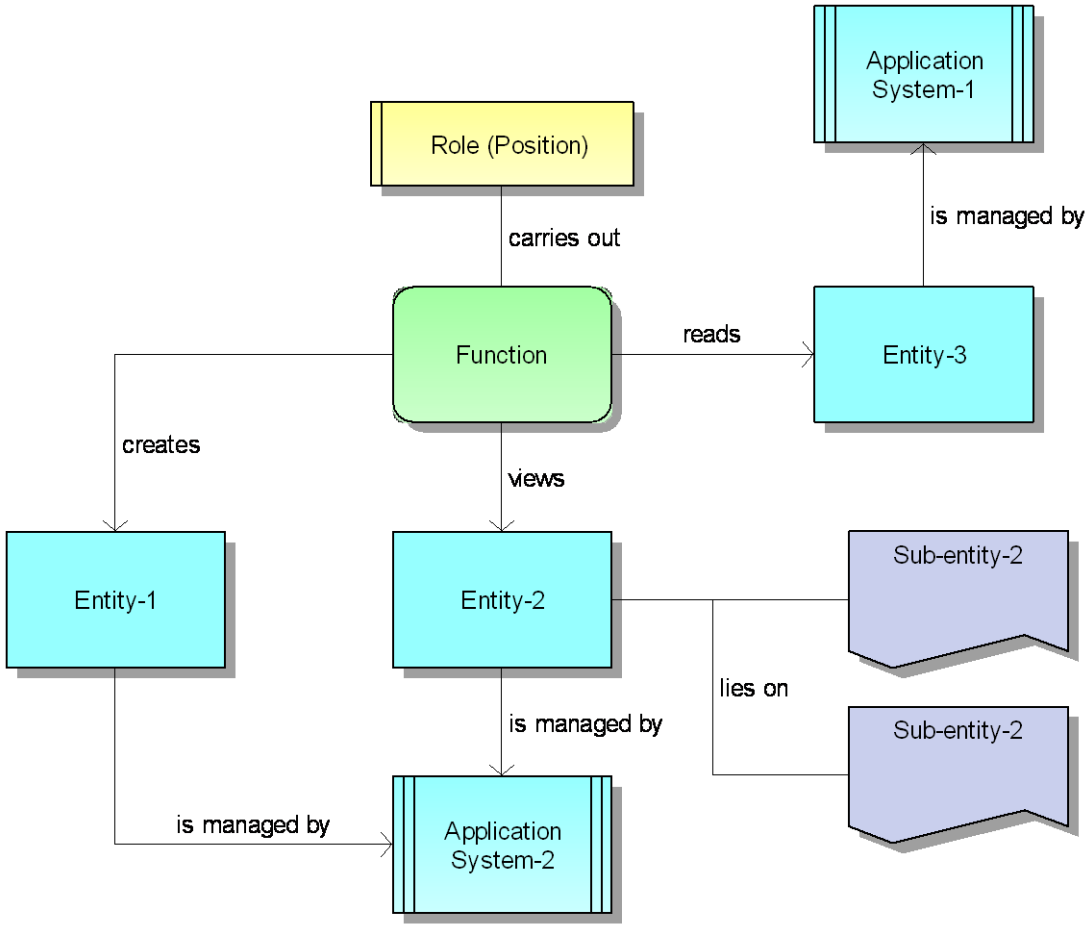


Figure 7: Instance of the function allocation diagram

Function of the FAD represents base functional components of the proposed method. Each relationship type represents a connection type such as *creates*, *reads* and *views* between a function and an entity. In addition, each entity has to be associated with an application system. Role represents user perspective of the model. For instance, we can interpret part of this diagram as “role views Entity-2 on Application System-2”. In addition, Entity-2 is composed of two sub-entities.

3.2. The Measurement Procedure

The aim of this measurement procedure is to represent mapping rules between business process model elements and COSMIC FSM method in order to construct e-Cosmic FSE method and its automation principles. Functional requirements are specified by means of business process flow for a product that is intended to support that business process.

Step 1. Identification of the Software Boundary

Software size varies in terms of identification of the software boundary, since the total size of software aggregated from separately measured software components is more than the size of measurement of whole software. The measurement of the software component but not the whole software system adds an entry-exit pair to the size of software during the interaction with other part of the software components. In addition to this, it is important to identify the users of the system. A role in the business process model or eEPC measurement model approach can be a person who is the user of the application or an external application and device. As a result, user concept of e-Cosmic FSE method is related to roles of eEPC measurement model. We have designed measurement procedure for e-Cosmic FSE method as follow;

- We should accept each role of eEPC measurement model as a user of the system.
- The border between roles of processes and the set of process flow diagrams can be accepted as the boundary of the system to be measured.

Step 2. Identification of Functional Processes

The main advantage of the business process modeling based on eEPC is event triggering dependency. Each function in the eEPC model has to be initiated by at least an event sent

by an actor or by the occurrence of a temporal event. Therefore, each function of the eEPC measurement model is identified as a functional process of e-Cosmic FSE method. The important part here is the level of granularity. We assume that each function that is identified as a functional process has to go down at FAD level, namely, this function does not include any business process flow assignment. COSMIC FSM method defines each functional process to be constituted of at least two data movements (Entry-eXit or Entry-Write), which has to be taken into account for definition of each functional process.

- We should accept each function of eEPC without any assignment as a functional process of e-Cosmic FSE method. It is mandatory to have a FAD diagram for each function involved measurement procedure.

Step 3. Identification of Object of Interest and Data Groups

The concept of data group is related to the object of interest. Moreover, COSMIC FSM method has not claimed definition of data group attributes as a mandatory phase for size estimation. However, the attributes of an entity can be accepted as attributes that must represent the smallest piece of information. Thus, entities which are specified in the FAD are identified as data groups because they represent a set of attributes of persistent objects. The roles are also candidate for entity so that they have a set of attributes. As a result, we have to add two new mapping rules related to the identification of data groups.

- We should accept each entity in the FADs as a data group.
- We should accept each role that participates in a FAD as a data group

Step 4. Identification of Data Movements

COSMIC FSM method is based on data movements that are Entry, Read, Write, and eXit. Data movements are identified in the relationship types represented in the function allocation diagrams. First of all, we have to consider output relationship types. Create, view, and list are basic output relationship types. As shown in Table 8 and Table 9, the appropriate set of data movement types are assigned to each output relationship types. The basic relationship types that are labeled with the above stereotype define data movement types in Table 8.

Table 8: Data movements of output relationship types

Output relationship type (stereotype)	Data movement types
Creates	1 Entry, 1 Write
Views	1 Entry, 1 Read, 1 eXit
Lists	1 Entry, 1 Read, 1 eXit
Changes	1 Entry, 1 Read, 1 eXit, 1 Write
Deletes	1 Entry, 1 Write
Archives	1 Entry, 1 Write
Updates	1 Entry, 1 Write

Table 9: Data movements of input relationship types

Input relationship type (stereotype)	Data movement types
Reads	1 Read
Views	1 Entry, 1 Read, 1 eXit
Uses	1 Entry, 1 Read, 1 eXit, 1 write

When an entity used according to these data movements includes attributes of different sub-entities, it is necessary to identify additionally each data movement for each sub-entity. The difference between *list* and *view* relationship types is that *list* relationship type is used for retrieving all instances of an entity in order to allow further selections whereas *view* relationship type is used for retrieving details of an entity. All confirmation and error messages within the scope of the function have to be accepted as only one eXit data movement.

3.2.1. Functional Size Calculation Algorithm

The purpose of this phase is to represent how the measurement procedure proposed in the previous subsection can be applied to the project consisting of business process model and to produce a quantitative value that shows the functional size of the software product. We

can apply that measurement procedure to the business processes either automatically or manually. Furthermore, a script was developed to estimate software size from the business process model automatically.

In order to define the size of the software product, firstly requirements of the software product presented as business process models has to be transformed to the functional user requirements. However, extracting the e-Cosmic FSE method elements from the functional user requirements is not only time consuming but also error prone work. Instead, elements of the e-Cosmic FSE method developed according to the COSMIC FSM method compatible with the business process model results in effective and correct deliverables than previously discussed approaches. The requirements in the business process model are detailed system-level requirements. The main measurement procedure of the e-Cosmic FSE method is depicted in Figure 8.

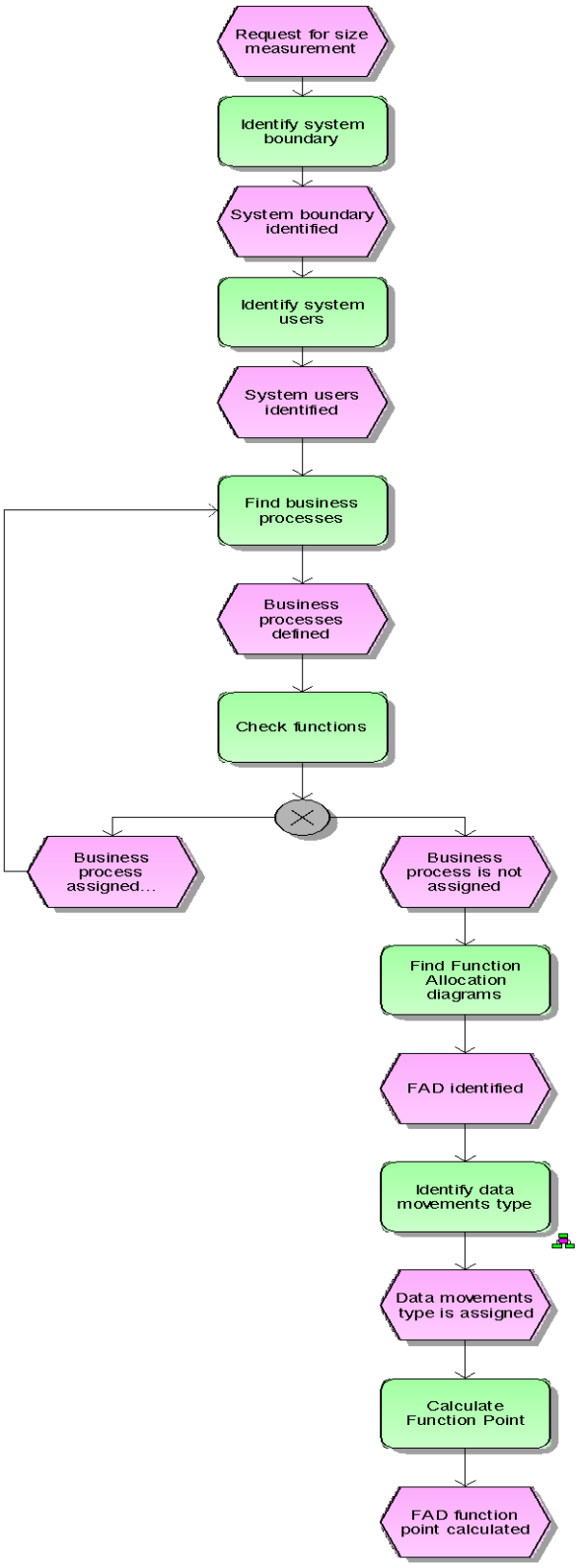


Figure 8: The main procedure of the e-Cosmic FSE method

The detail of the Identify Data Movement Type step is depicted in Figure 9.

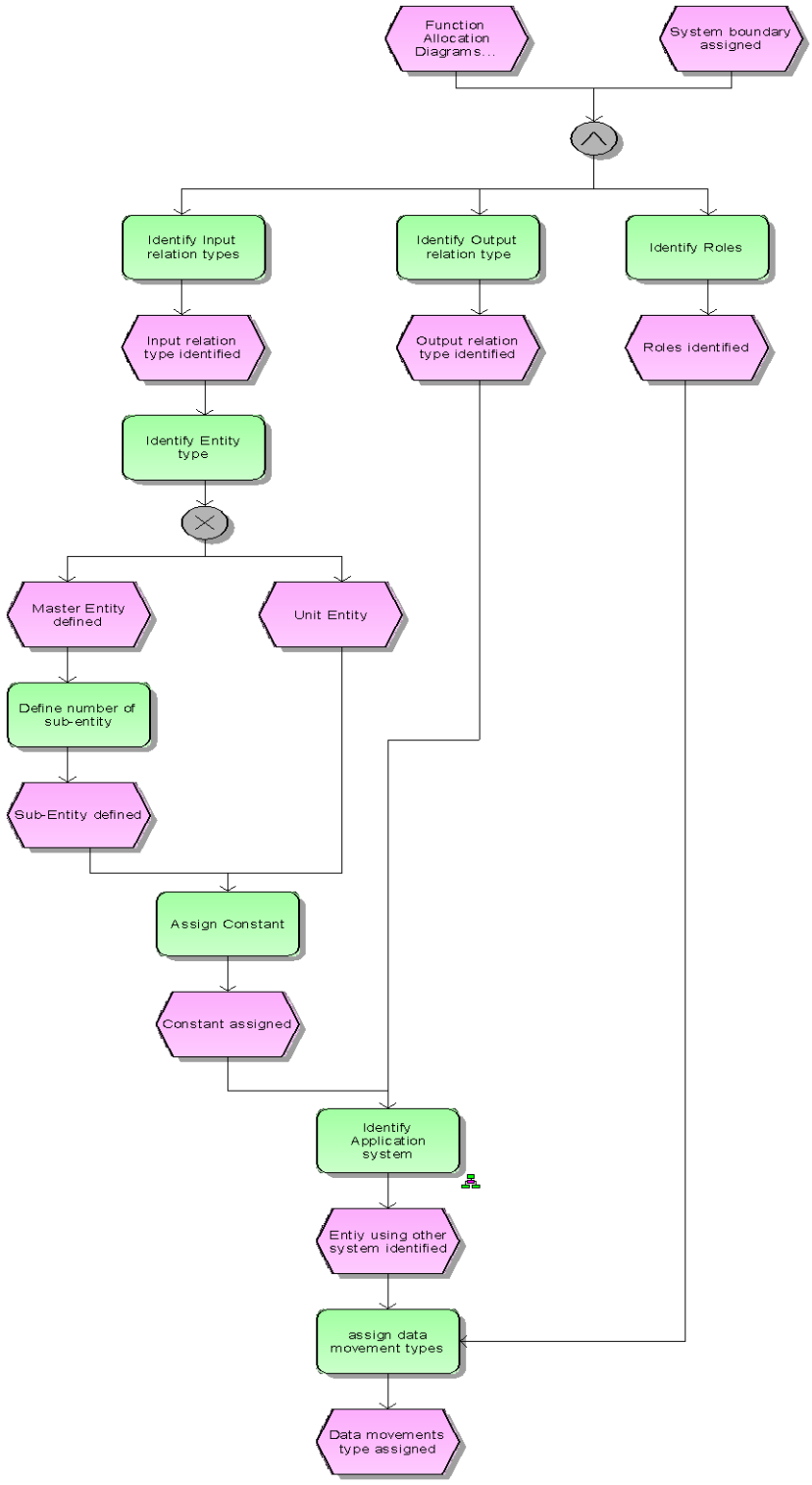


Figure 9: The detail of the Identify Data Movement Type Step

The detail of the Identify Application System step is depicted in Figure 10.

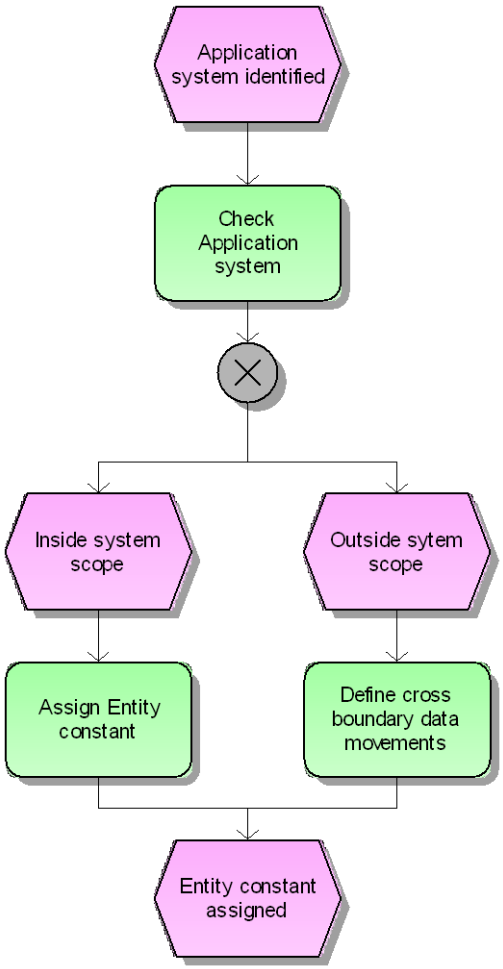


Figure 10: The detail of the Identify Application System step

According to algorithm defined in Figure 8, Figure 9 and Figure 10, firstly software boundary has to be identified as either a whole database that business process model is based on for structure or a module that is a part of database. As a result of this definition, interactions between users and system are easily defined. Then, business processes generating the modules are identified as a starting point of the function definition. The most important part of the function definition is to determine it without any business process assignment. Functions without any business process assignment are named as base functions from this point.

Base function is figured as FAD in order to determine input, output, relationship types, roles and application systems. Input and output relationship types definition is the most important part of measurement in terms of assignment of data movements

Application system type definition help the determination of the boundary and application systems that are placed outside the software measurement scope also connect some entities. The interaction of those type entities with measured system requires entry-exit pair of data movements. Entities are also separated into two parts; master entity and unit entity. This separation not only affects software size but makes the proposed model more suitable to the COSMIC FSM method with respect to identification of objects of interest as well. Since Master Entity consists of more than one sub-entity, we have to add set of data movements as many as the quantity of sub-entities to the function point of function allocation diagram.

Input and output relationship definition determines each entity behavior since each entity is assigned a set of data movements in terms of relationship types. The assignment of a set of data movements is explicitly identified in previous subsection.

The functional size of a FAD is equal to the sum of all data movements defined in the same FAD. The size of whole system is found by aggregation of functional size of functional allocation diagrams.

3.3. Method Implementation

Business process model is developed in ARIS tool environment; therefore, same environment also is used for automating the functional size of the software product. The tool has been developed by IDS Scheer AG to support consultants and companies in creating, analyzing, and evaluating company processes in terms of business process (Scheer, 2001). Each object of the ARIS tool has lots of attributes that consist of common and objects specific properties such as Name, Identifier, and Description. The ARIS toolset enables users to write their own scripts or edit written scripts through a script editor.

The e-Cosmic script was developed using the ARIS Scripting language which is based on the JavaScript programming language. ARIS Script is a scripting language of ARIS Tool. The core functionality of the e-Cosmic script reports functional size of the software product from business process model. In addition, functional size generation can be filtered according to the groups of the database in order to report partial size of the software.

The execution of the script starts from taking necessary information such as language, database, group, output format and model for the script and controls them. Firstly, database or group selection is completed according to desired functional size measurement approach. And then all groups selected before is searched and sorted by group hierarchy. When these processes are completed, function allocation diagrams are constructed and its functions are identified. Since each function is composed of roles, entities and application systems, counting procedure starts from identifying each of them. All these processes are executed according to the algorithm described in the previous section.

ARIS tool scripting language is composed of various predefined classes and methods. The script is not only composed of sub programs but uses these predefined classes and methods as well. The summary of ARIS tool classes and methods are described in the following subsections (ARIS tool, 2009).

3.3.1 Description of Classes

The classes described below are the defined classes of the ARIS Tool Set which are used in the tool. In addition, only common classes used in the script are mentioned below.

1. **Context object:** The object is a basic class of ARIS tool used to define language and output format.

   getSelectedLanguage ( ) : Returns the selected database language

   getSelectedFile ( ) : Returns the name of the output file. If several output files have been created, this is the name of the file displayed after running the report.

   getSelectedFormat ( ) : Returns the selected output format

2. **ArisData Object :** The ARIS start object is registered as the "ArisData" object in each script. It provides database-dependent methods and objects, e.g. the selection data for user (selected groups, models, objects) in the script.

   getSelectedFilters ( ) : Returns the list of selected method filters (in context) as a list of MethodFilter objects.

   getSelectedDatabases ( ) : Returns the list of selected databases (in context) as a list of Database objects. In the current version only one database can be selected. The list contains only the database that is also returned by ActiveDatabase.

   getSelectedGroups ( ): Returns all selected groups. Navigation and inquiries are carried out using the ARIS object model.

   sort (Object[] aObjects, int kriterium1, int localeID) : Sorts the transferred array containing items of the "Item" class (objects, models etc.) based on the specified sorting criteria.

3. **Database Object**

   Find (int SearchKind): General database search. Return all items in the database that correspond to the specified search criteria.

4. **Group Object**

   ModelListFilter (int modelTypeNum) : Returns a list of all models in the group with the specified model type.

5. **Model Object**

   ObjOccListFilter (int value) : Returns the object occurrences in the model which meet the specified criteria as a list of object occurrences

6. **ObjOcc Object**

   ObjDef ( ) : Returns the object definition as ObjDef .

   CxnOccList ( ) : Returns all relationship occurrences attached to the object occurrence.

7. **CxnOcc Object:** The object represents a relationship occurrence.

   CxnDef ( ) : Returns the relationship definition.

   ActiveType ( ) : Returns the active name of the relationship.

SourceObjOcc ( ) : Returns the occurrence of the source object.

TargetObjOcc ( ) : Returns the occurrence of the target object.

8. **ObjDef Object**

   ActiveType ( ) : Returns the active name of the relationship.

9. **Item Object:** This class contains the common methods of all attribute-bearing items and is also the basic class of all specialized items (object, model, etc.)

   Name (int): Returns the name of the item in the specified language as a string.

10. **Output Object:** The object is an Output object for the ARIS report and outputting documents in RTF, HTML, XLS, PDF and Text (UTF-8) formats.

    OutputLn (String p_sText, String p_Font, int p_iFontSize, int p_FontColor, int p_BkColor, int p_Format, int p_iIndent): Output of a text in report with line break.

CHAPTER 4

CASE STUDY

This chapter presents case studies performed to validate e-Cosmic Functional size estimation method.

4.1 Overview

We performed case studies in order to analyze functional size measurement potential of business process models and validate measurement results.

E-Cosmic FSE method requires business process model as input artifacts to apply its measurement procedure. We have applied the proposed method to a continuing project. We explore the applicability of business process models for functional size measurement and developing an approach for doing this during the first case study. The second and third case studies are performed for comparison and method validation.

4.2 Case Study Design

We planned three case studies. The goal of the first case study was to explore applicability of the proposed method. Hence, use cases of the first case study were created in order to apply COSMIC FSM method for finding size of the first software product. When e-Cosmic FSE method was applied to the business process model manually to quantify software size of the first product, we also examined possible deviations from the size obtained by

applying COSMIC FSM. Because of these concerns the first case study was not used for validation purposes.

Second case study was conducted with a wider view of the applicability of the method. Although both cases are in the information system domain, they reflect different views on software products. The second and third cases are more complex data-driven functions than the first case study. In addition, second and third case studies also contain measurement result of other early estimation method in order to compare their results to the results of the proposed estimation method. Third case study was performed after the script was applied to the whole project. According to the algorithm presented at the previous chapter, script was executed to take whole project as an input. When whole project functional size was determined, we decided to implement the third case study in order to complete the validation purpose. All of the selected case products are in the MIS domain.

The details of the case studies are described in this chapter. The main aim of the experimental study was to validate the e-Cosmic FSE procedure. Other aims were to determine possible improvement opportunities and measure the effectiveness of the method and discover if there were other benefits of the method. The aim of this thesis study is not only to create an early functional size estimation method and support it with measurement manual but to automate this measurement procedure as well.

The research questions of the case studies are

- What are the business process model's elements that can be utilized for measuring functional size?
- What are the problems and difficulties of the current early functional size estimation based methods and improvement opportunities?
- Is there a significant deviation between the proposed early size estimation method and COSMIC FSM method results?

- How much does the proposed early size estimation method improve software size estimation with respect to the available methods in the literature?

The implementation plan of the case studies is depicted in Table 10.

Table 10: Implementation plan of Case Studies

No	Activity to be performed
	Case study 1 (Part 1)
1	Identification of a coherent set of business processes.
2	Software requirement specifications of the first set of business processes are constituted.
3	The size of the first set of business processes is counted based on defined requirements by using COSMIC FSM v3.0.
4	e-Cosmic FSE developed in the context of this thesis study is applied to count the size of the first set of business processes.
5	e-Cosmic FSE v1.0 is constituted.
	Case Study 2 (Part 2)
6	Identification of a coherent set of business processes.
7	Software requirement specifications of the second set of business processes are constituted.
8	The size of the second set of business processes is counted based on defined requirements by using COSMIC FSM v3.0.
9	e-Cosmic FSE is applied to count the size of the second set of business processes.
10	Other early estimation method (Early and Quick COSMIC FFP) is applied to the first and second set of business processes in order to compare results.
	Case Study 3 (Part 3)
11	Identification of a coherent set of business processes.
12	Software requirement specifications of the third set of business processes are constituted.

Table 10 (continued)

13	The size of the third set of business processes is counted based on defined requirements by using COSMIC FSM v3.0.
14	Other early estimation method (Early and Quick COSMIC FFP) is applied to the third set of business processes in order to compare results.
15	Script is completed by using ARIS tool scripting language and applied all set of business processes.
16	Functional size of the third set of business processes is calculated by executing script.
17	Results are compared.

4.2.1 Case Study 1: Constructing e-Cosmic FSE method

The goal of the first case study is exploration of the similar concepts in business process models and functional size measurement method and establishment of a draft early size estimation method. Specifically, the first case attempts to address the research question of identifying the business process model elements that can be utilized for measuring functional size and point out the problems and difficulties of the current early functional size estimation based methods. We also explored any improvement opportunities during the early functional size estimation by means of comparing the proposed method results to the COSMIC FSM results.

Business processes of the “Extending Independent Auditor Pool” (EIAP) of Development Agency were chosen as the business process set to be applied. We selected EIAP since it displays an adequate level of complexity. That is, EIAP has different process levels which provide a rich environment to test the proposed method. But it is not too complex so that it is possible to identify improvement opportunities for e-Cosmic.

EIAP business processes were modeled with eEPC by using ARIS tool. We determined that eEPC business process models is suitable for identifying COSMIC base functional components, we also utilized a resource allocation mapping concept for each function

involved in the size measurement scope of the eEPC diagrams. We also decide whether function might be a part of the software product at this phase. If we develop a FAD for a function, we associate this function as a part of the software product.

We determined the business processes of the first case study that is part of the Development Agency Process Modelling project in order to count size of the software product. To calculate the size of the software product we need to determine the requirements of the same product. COSMIC FSM is designed to measure software product size from its FURs. Uses cases are used to extract FURs, and then each FUR is decomposed to functional processes.

We extracted the functional user requirements as input to the COSMIC FSM method for software size estimation. To provide the integrity among the size measurement results, the functional sizes of the case products measured were filled in a preformatted Excel table which consisted of “functional process name”, “data movement type”, and “data group description”. Then this size of the software product was verified by a certificated COSMIC FSM measurer. Use case models are also verified by an object-oriented analysis/design expert. In addition, all of the sizes of the case products were measured by using COSMIC v3.0 by the author of this thesis. The use case model and COSMIC FSM method results are given in Appendix B.

Once the size of the software products was measured, we started constructing e-Cosmic FSE procedure on the business process models of the first case study in order to estimate functional size manually.

The instance of business process model in the EIAP considered being a part of the software product is given in Appendix A. The business process model includes 9 functions initiated by events. Each function also includes 1 FAD to represent input-output relationships of resources. However, the level of granularity of one function of this business process is not the same as with the other functions; so a new business process is assigned to this function. This new business process also includes 6 functions initiated by events. Totally, two distinct

diagrams consisting of 14 functions were created to model existing business processes of organization unit by using the eEPC notation. The reason why these business processes are selected is to include opportunities for possible improvements. Firstly, measurement procedure is applied to the business processes, and then refinements have been identified through this measurement procedure.

In order to find any shortcomings during the method procedure, every step was monitored. The summary of the measurement results of the first case study using the e-Cosmic FSE method is given in Table 11 in the case study results part.

4.2.2 Case Study 2: Comparing Estimation Methods

Second case study is more related to the performance of the proposed e-Cosmic method. This case study addresses two research questions. The first question is whether there is a significant deviation between proposed early size estimation method and software size obtained by applying COSMIC FSM. Second question we ask is by how much software size estimation is improved by e-Cosmic method with respect to the available methods in the literature. In order to answer these questions, results from e-Cosmic FSE method are compared with another early estimation method; Early and Quick COSMIC FFP.

The second case study is performed on “Perform Project Payments” (PPP) business process models of Development Agency. This case study is deliberately chosen as a bigger boundary than first case study. The first case study included 14 functions whereas the second case study involves three distinct diagrams consisting of 29 functions to model existing business processes of organization unit by using the eEPC notation. This increase creates higher complexity and presents a better ground for comparing the performances of e-Cosmic to Early and Quick COSMIC FFP. The second case study consists of the PPP activities. Business processes of this case study contain arrangement of payment system for each project. The conduct of each payment phase such as prepayment, interim payment and final payment is handled. In addition, project contractor related issues are also solved in this phase.

The conduct of the second case study is the same as the first case study. Firstly, we determined requirement specifications. Uses cases were created to extract FURs so as to apply COSMIC FSM method procedure and then proposed early size estimation method was applied to find functional size of the software product. The case study result is explained in the case study results part.

The implementation of Early and Quick COSMIC FFP is that this method makes use of both analogical and analytical classification of functionalities (Meli and Santillo, 1999). The key factor defining detail level of the method is determined as function of eEPC model. Each function of the eEPC model is classified in terms of their complexity and a set of FP values (min, avg, max) based on analytical tables are assigned to each object. The sum of values generates estimate of software product size.

4.2.3 Case Study 3: Implementation of e-Cosmic FSE method

The aim of the third case study is quite similar to the purpose of conducting the second case study. Same research questions are addressed. Firstly, we investigate the existence of deviation between proposed early size estimation method and software size obtained by applying COSMIC FSM. Secondly, we try to assess the software size estimation improvements brought about by the proposed early size estimation method. This second research question is tackled by comparisons to the available methods in the literature. Even though the research questions of the two case studies are identical, the third case study involves automation of the proposed e-Cosmic method.

When we decided to select the third and last case study, we have attempted finishing the script which executes the automation procedure of the e-Cosmic FSE method. After finishing the script of the e-Cosmic FSE method in the ARIS tool environment, we determined business processes of the third case study as input for script to estimate functional size of the last case study. We also determined requirements. Uses cases were created to extract FURs so as to apply COSMIC FSM method procedure.

In order to answer the formulated research questions, the written script is executed and obtained results are compared with the result of the software products obtained by applying COSMIC FSM. Thus, with the last case study, we validate not only the result of the measurement procedure but the result of the automation procedure as well. The last case study is of nearly the same boundary size as the second case study.

The last case study consists of the "Problem Management" (PM) activities. Early warning, infraction of rules and contract termination works are conducted in the last case study. The third case study involves three distinct diagrams consisting of 29 functions to model existing business processes of organization unit. This case, as in the second case study, provides a complex environment to compare results of different methods. It is also complex enough to challenge an automated software size estimation method. The last case study result is explained in the case study results part.

4.3 General Discussions on the Case Studies and the Results

The first case study is deliberately more detailed than other case studies for we want to increase the comprehensibility of the steps of the e-Cosmic measurement procedure and then the case study results and our inferences based on these results are explained in the following parts. The estimation results of the "Extending Independent Auditor Pool" business processes are given in Table 11.

Table 11: Summary of e-Cosmic FP of the "Extending Independent Auditor Pool"

Functional Process	# of Entry	# of Exit	# of Read	# of Write	CFP count
Create IA selection criteria	3	3	3	1	10
Create IA advertisement text	3	1	1	1	6
Create IA reference documents	1	0	0	1	2
Plan IA advertisement time, advertisement tool, and application time	1	0	0	1	2

Table 11 (Continued)

Send outputs to approve	2	1	1	1	5
Review IA outputs	1	5	5	1	12
Update IA outputs	5	5	5	4	19
Prepare for advertisement tools	1	0	1	1	3
Announce acquisition advertisement	1	4	4	0	9
Receive IA applications	2	0	1	1	4
Determine selection committee	2	1	1	1	5
Select IA applications	5	2	2	4	13
Approve selected IA	3	2	2	2	9
Save selected IA	2	0	1	1	4
List IA selection criteria	1	1	1	0	3
Update IA selection criteria	1	0	0	1	2
Delete IA selection criteria	1	0	0	1	2
List IA advertisement text	1	1	1	0	3
Update IA advertisement text	1	0	0	1	2
Delete IA advertisement text	1	0	0	1	2
List IA reference documents	1	1	1	0	3
Update IA reference documents	1	0	0	1	2
Delete reference documents	1	0	0	1	2
List IA acquisition plan	1	1	1	0	3
Update IA acquisition plan	1	0	0	1	2
Delete IA acquisition plan	1	0	0	1	2
				TOTAL CFP	131

The results obtained from the COSMIC FSM method are represented for each functional process of the business processes and data movements are also represented for each function in Table 12.

Table 12: Cosmic Function Point of "Extending Independent Auditor Pool" by means of COSMIC

Functional Process	# of Entry	# of Exit	# of Read	# of Write	CFP count
Create IA selection criteria	3	3	3	1	10
List IA selection criteria	1	1	1	0	3
Update IA selection criteria	1	0	0	1	2
Delete IA selection criteria	1	0	0	1	2
Create IA advertisement text	3	**2**	1	1	7
List IA advertisement text	1	1	1	0	3
Update IA advertisement text	1	0	0	1	2
Delete IA advertisement text	1	0	0	1	2
Create IA reference documents	**2**	1	1	1	5
List IA reference documents	1	1	1	0	3
Update IA reference documents	1	0	0	1	2
Delete reference documents	1	0	0	1	2
Plan IA advertisement time, advertisement tool, and application time	1	0	0	1	2
List IA acquisition plan	1	1	1	0	3
Update IA acquisition plan	1	0	0	1	2
Delete IA acquisition plan	1	0	0	1	2
Send outputs to approve	2	1	1	1	5
Review IA outputs	1	**6**	**6**	1	14
Update IA outputs	5	5	5	4	19
Prepare for advertisement tools	1	0	1	1	3
Announce acquisition advertisement	1	**5**	**5**	0	11
Receive IA applications	2	0	1	1	4
Determine selection committee	2	1	1	1	5
Select IA applications	5	2	2	4	13
Approve and save selected IA	4	2	2	3	11
			TOTAL CFP		137

When the COSMIC functional processes which are extracted from the requirement specifications depicted in Table 12 are compared with the functional processes of the e-Cosmic FSE procedure in Table 11, we find out few differences. Also, the quantity of functions of the use case model is one function less than the quantity of proposed measurement procedure functions. The difference resulted from the combination of two functions as one function in COSMIC FSM approach. Since identification of use cases is left to the measurer, such deviations may occur in the COSMIC FSM method, legitimizing the concern for subjectivity. Still, software product size estimation from the use case brought about few refinement potentials on the proposed measurement procedure. These improvements are listed below.

- User interactions have to be taken into account. For this purpose we revised the *view* relationship type. The differences of *view* and *read* relationship types are not only quantity of data movements but their measurement view point differs as well since *read* does not take into account any user interaction. On the other hand *view* relationship type is identified to reflect user interactions better.
- For each function, we added one triggering entry in order to consider user role effects.
- We also clarify effect of sub-entity type; if an entity lies on sub-entities, we have to increase quantity of data movements depending on quantity of sub-entities.

The other improvement is to find that *create* relationship requires *list*, *delete* and *update* relationships for the size estimation. When we need to use *create* relationships, an entity is created, and so each created entity would be updated and deleted by users. To perform this operation, we also require *list* relationship type for each entity. *Update*, *delete* and *list* data movements pattern is given in Table 13 below.

In Table 13, data movements that are allowed in the relationship type are marked with an * sign.

Table 13: Update, delete, and list pattern for an entity

Relationships type	Entry	Exit	Read	Write
Update	*			*

Table 13 (Continued)

Delete	*			*
List	*	*	*	

The business processes of the software products was calculated by means of this pattern for four functions that require *create* relationship. Therefore, software product size would be 131 FP. This increase is inevitable for a business process model because of the addition of the size of *list*, *delete* and *update* operations.

When the results obtained from e-Cosmic which is 131 CFP and COSMIC FSM (Table 12) which is 137 are compared, it is found that the difference between the results is 6 CFP. The software product size estimation from the business process model for this particular example resulted in 5% lower result than the COSMIC FSM value that was produced by using the use case model. The functional processes where differences from Table 11 occurred are represented with bold numbers in Table 12.

The reason for such a difference is that the measurement by using use case model considers triggering event for each function that starts without any triggering event. However, this difference is not based on COSMIC measurement procedure as a special rule. This variance purely resulted from measurer's interpretations of the FUR. The other differences also reflect measurer's interpretations. For instance, when a function completes its task, the last purpose of such a function either saves results or shows results to the user. However, if we want to do both operations for some functions, these operations also increase data movements and contribute to the size of software product.

We continue explaining relationship types and measurement procedure by giving two function allocation diagrams in order to clarify steps. The first function of the EIAP business process, "*create IA selection criteria*", depicts relationships and sub-entity situations (Figure 11).

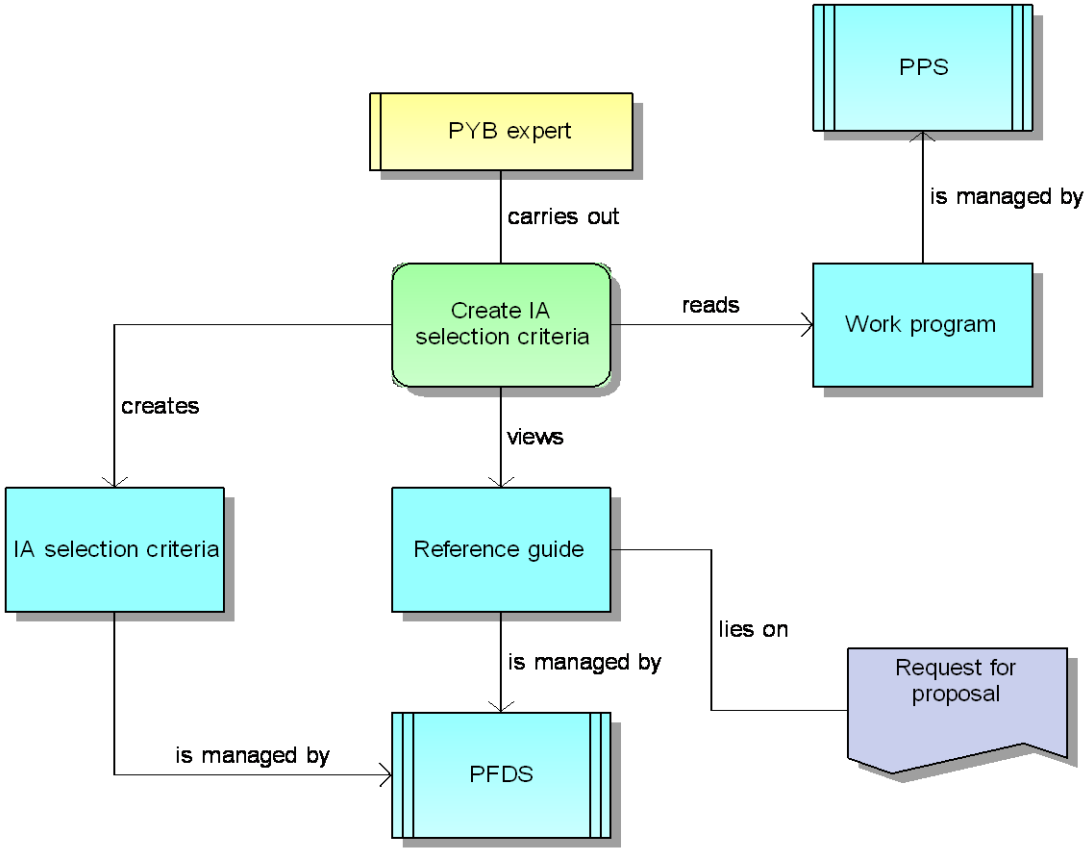


Figure 11: Function allocation diagram of "create IA selection criteria"

In Table 14, data movements that are allowed in the relationship type are marked with an * sign.

Table 14: Mapping of relationship types into data movements for "create IA selection criteria"

Relationships Type	Entry	Exit	Read	Write
Reads			*	
Views	*	*	*	
Creates	*			*

The function depicted in Figure 11 includes two application systems; however, both of them are in the same measurement scope. Therefore, we do not have to take into account the application system so that there are not any effects to increase the data movements.

The other important part is sub-entity type. “Reference guide” entity includes request for proposal sub-entity. So according to proposed measurement procedure, we have to add data movements of “request for proposal” for each set of data movements of “reference guide”. We can also see roles relationship types as “carries out”. The second function of the EIAP business process, “*update IA outputs*”, is given in the detail in Figure 12 to depict measurement procedure steps.

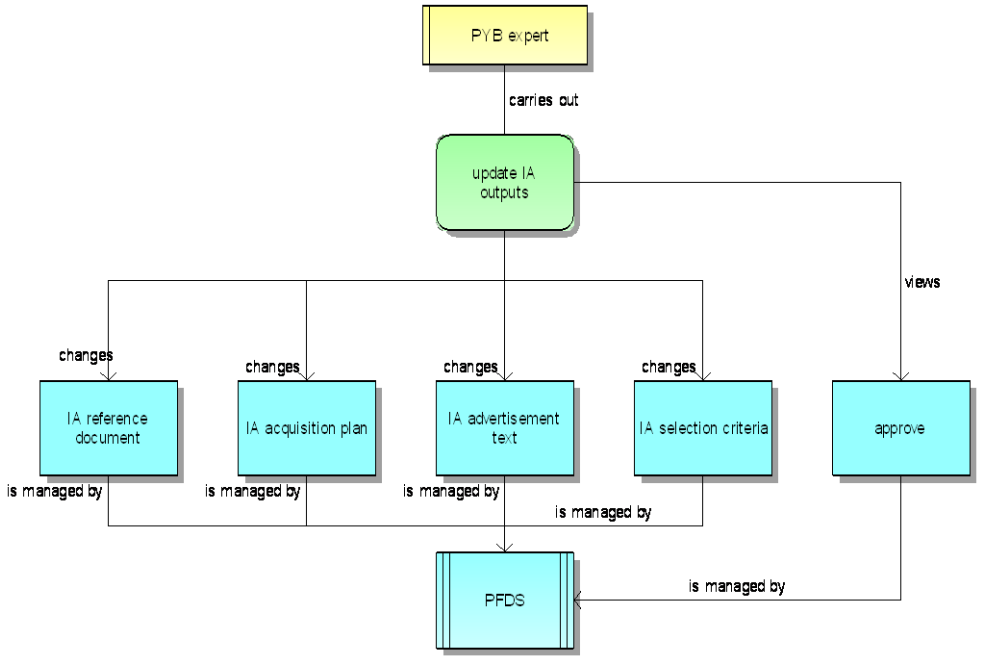


Figure 12: Function allocation diagram of “update IA outputs”

Table 15: Mapping of relationship types into data movements for “update IA outputs”

Relationships	Entry	Exit	Read	Write
Views	*	*	*	
Changes	*	*	*	*

The function in Figure 12 explains an important part of measurement procedure in terms of the *change* relationship types. *Change* relationship type includes both *update* and *list-before update* data movements. Also, *view* relationship type differs from *read* with respect

to quantity of data movements. User interactions are classified as *list*, *list-before update*, and *list-before create*. *View* relationships type covers all of them.

Functional sizes are computed by the e-Cosmic method for each functional process separately. FP count values of each software product are given in Table 16. This table provides data for a statistical analysis.

Table 16: CFP count of each functional process for software products by e-Cosmic

Functional Process	CS-1 (CFP Count)	CS-2 (CFP Count)	CS-3 (CFP Count)
fp-1	10	25	6
fp-2	7	7	15
fp-3	6	12	7
fp-4	7	7	7
fp-5	2	4	6
fp-6	7	6	8
fp-7	2	6	7
fp-8	7	7	8
fp-9	5	7	15
fp-10	12	6	6
fp-11	19	22	7
fp-12	3	14	7
fp-13	9	9	15
fp-14	4	7	7
fp-15	5	16	3
fp-16	13	7	6
fp-17	9	12	3
fp-18	4	7	10
fp-19	-	6	7
fp-20	-	24	7

Table 16 (continued)

fp-21	-	7	3
fp-22	-	8	3
fp-23	-	24	5
fp-24	-	7	15
fp-25	-	12	3
fp-26	-	7	7
fp-27	-	9	10
fp-28	-	6	8
fp-29	-	7	3

Based on the data available in Table 16 and Table 17 summarizes the mean and standard deviation of the functional processes of three software products. Software product in case Study 2 has the highest mean and standard deviation values with respect to functional size of each functional process. Comparing columns 1 and 2 in Table 17 shows that the average functional size of the functional processes and the related standard deviation increase with the number of functional processes of the software product. However, comparing columns 1 and 3 shows that while the average functional size of the functional processes may remain unchanged, standard deviation may even decrease as the number of functional processes of the software product increases. Finally, columns 2 and 3 taken together imply that the average functional size of the functional processes and the related standard deviation may be independent of the number of functional processes of the software product.

Table 17: Statistical analysis of the case studies with respect to size of the functional processes

	CS-1	**CS-2**	**CS-3**
Sum (CFP)	131	298	214
Mean	7.28	10.28	7.38
Std	4.30	6.11	3.67

Table 18 shows that estimation deviations of the e-Cosmic FSE method from the COSMIC FSM method for software products are 4.38%, 6.29% and 5.31% respectively. As shown in Figure 13, the deviation of results obtained from e-Cosmic FSE method with respect to COSMIC FSM method are seen to increase with the increase in the functional size of the software products. Also, estimation deviations are acceptable in order to validate the e-Cosmic FSE method according to the other early functional size estimation ranges. There are not any differences between quantity of functional sizes of the e-Cosmic FSE and COSMIC FSM. We have not observed any effect of the quantity of the functional processes on the functional sizes clearly.

Table 18: Deviation between total functional sizes of the software products with respect to the methods

	CS-1	CS-2	CS-3
E-Cosmic FSE (Total CFP)	131	298	214
COSMIC FSM (Total CFP)	137	318	226
Estimation deviation(CFP)	6	20	12
% Estimation deviation	4.38	6.29	5.31
# of Functional Process	18	29	29

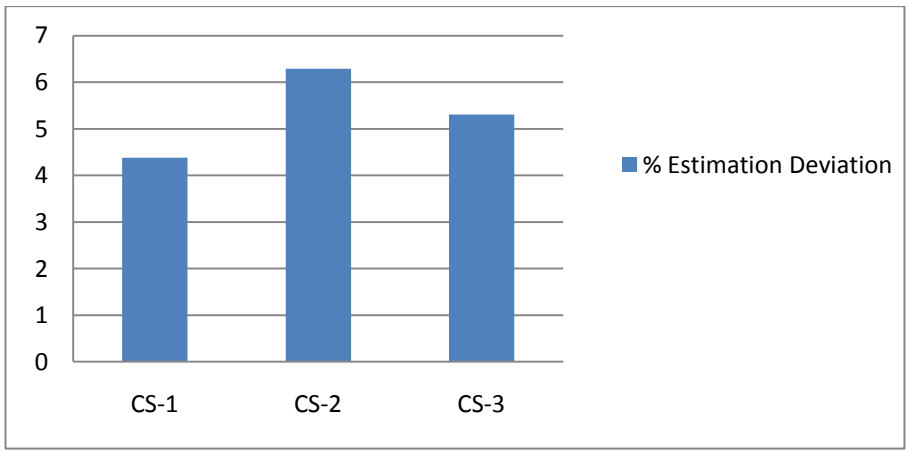


Figure 13: Estimation deviations between total functional sizes of the software products with respect to the methods

Early and Quick COSMIC FFP method, another functional size estimation method, was applied to the three case studies according to the values based on analytical tables as

standard by Early and Quick COSMIC method's developers. Results are given in Table 19. We have used values published by Conte et al. (2004).

Table 19: Implementation of Early and Quick COSMIC FFP method on the case studies

	Min. CFP	Most likely CFP	Max. CFP
CS-1	123.00	190.80	241.00
CS-2	223.00	345.30	438.00
CS-3	178.00	275.70	348.00

Table 20: Size estimates by Early and Quick COSMIC FFP and the deviations with respect to COSMIC FSM.

	Min.	Most likely	Max.
CS-1	-10.22	39.27	75.91
CS-2	-29.87	8.58	37.74
CS-3	-21.24	21.68	53.98

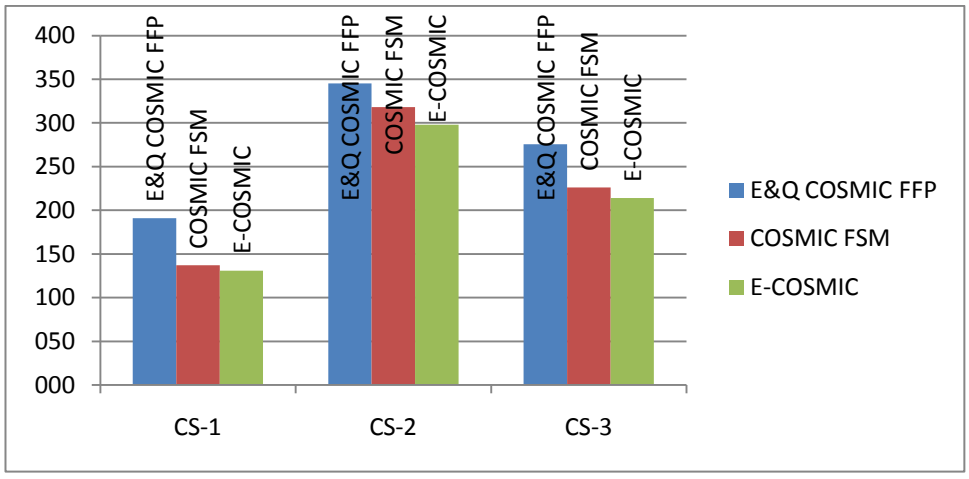


Figure 14: Comparison of the functional size of the software products with respect to the employed methods

As shown in Figure 14, if we compare the functional size of e-Cosmic FSE method with results of COSMIC FSM method, we can observe that functional sizes are underestimated

by using e-Cosmic FSE. However, the differences are not significant if one considers 10% as a deviation benchmark for significance, a benchmark proposed by Santillo and Meli (1998). In addition, the functional sizes are overestimated by using the E&Q COSMIC FFP 2.0 (Conte et al., 2004). We used average values of the E&Q COSMIC FFP 2.0 method for comparison in Table 20. The results of the E&Q COSMIC FFP 2.0 are not so reliable according to the results of the other two methods. On the other hand, upon comparing e-Cosmic and E&Q COSMIC FFP 2.0 values with COSMIC FSM values, we can see that e-Cosmic values are closer to COSMIC FSM values than E&Q COSMIC FFP 2.0 values.

Effort values of COSMIC FSM and e-Cosmic FSE method are summarized in Table 21. As shown in Table 21, the effort needed to make an e-Cosmic estimation is found to be about 44% less than the effort to make a measurement with COSMIC FSM for the software products. We can conclude that e-Cosmic FSE method reduces the required effort for measurement.

Table 21: Effort values for COSMIC FSM and e-Cosmic

		Case Studies		
Method	**Phase**	**CS-1**	**CS-2**	**CS-3**
COSMIC FSM	Requirement Specification (man-hours)	14	10	10
	Size Measurement (man-hours)	10	9	9
e-Cosmic	Size Measurement (man-hours)	6	5	5

In summary, compared to a benchmark case of COSMIC FSM method, e-Cosmic performs at least no worse than the E&Q COSMIC FFP method. e-Cosmic method differs from E&Q COSMIC FFP method in determination of components to be used in software size measurement. e-Cosmic is based on business process models, a relatively well formulated framework that is less prone to deviations caused by subjectivity. E&Q COSMIC FFP

method, however, lacks such a foundation. Therefore, the analysis conducted so far implies non-negligible gains from formalization of software size estimation methods.

CHAPTER 5

CONCLUSION AND FUTURE WORK

This chapter summarizes results, contribution, and significance of this research and suggests future research directions.

5.1. Conclusion

In this study, a method of early functional size estimation based on Business Process Models is developed. Business process model artifacts, which are widely used in information system environment, are used as input to e-Cosmic FSE method.

We also developed a software that automates parts of the methodology. In order to construct and verify the applicability of e-Cosmic method, we performed three case studies. COSMIC FSM method, E&Q COSMIC FFP and e-Cosmic FSE methods were applied during these case studies in order to find the functional size. The improvement opportunities have been observed by comparing the results.

Based on the three case studies conducted we have found out that e-Cosmic FSE method can successfully estimate the size of the software to be developed. Santillo and Meli (1998) suggested 10% deviation benchmark for significance. Our results have consistently displayed less than 10% deviation. The estimation deviations vary from 4.38% to 6.29% with respect to COSMIC FSM results. These deviations resulted from additionally identified *exit* data movements in the COSMIC FSM method in order to display function

results to the users and the *list* relationships. E-Cosmic FSE method underestimated the software product size.

The estimation deviations of E&Q COSMIC FFP vary from 8.58% to 39.27% with respect to COSMIC FSM results. When we compare estimation deviations of E&Q COSMIC FFP and e-Cosmic FSE methods values for each case study, we can see that the estimation deviations of e-Cosmic FSE are significantly smaller than the estimation deviations of the E&Q COSMIC FFP. Based on these comparisons, we can conclude that e-Cosmic FSE method produced more precise result than the other method in the literature.

In addition, since the reliability of the E&Q COSMIC FFP is directly proportional to the estimator's ability to "recognize" the components of the system as part of one of the proposed classes, E&Q COSMIC FFP method entails a large degree of subjectivity. We have largely eliminated the subjectivity by e-Cosmic FSE by precisely defining the counting rules based on the process models. This study showed that business process models can be used for decreasing subjectivity and increasing reliability of the early size estimation technique.

Conventional FSM methods accomplished automation of the measurement procedure based on software model artifacts at a late phase of the software life cycle. We also show that e-Cosmic method provides software size estimation at an early phase of the software life cycle when business process model is available. Business process model artifacts are chosen due to the fact that it is suitable for formalization of the user requirements. They are widely used in information system domain and available at the beginning of the software life cycle.

Another result is that the effort needed to make e-Cosmic FSE is less than the effort to make E&Q COSMIC FFP and COSMIC FSM. The effort needed to make manual calculation by using COSMIC FSM was 9 man-hours. On the other hand, the effort needed to make estimation based on e-Cosmic FSE was a few man-minutes for the same product. This reduction in man-hours is due to automation of the size estimation process in e-Cosmic FSE.

The constructed algorithm enabled automation of the size measurement process. The automation, in turn, led to improvements in effort needed for size estimation process.

The automation of the measurement procedure eliminates measurer dependency and manual calculation effort in conventional early functional size estimation methods. This also decreases advanced measurement training needs.

5.2. Future Work

E-Cosmic FSE method can be applied to different application domains. The performed case studies were in management information systems. In order to make method more general, we should apply the method to different functional domains such as real-time systems or control strong systems. We also should apply method to more case studies.

This study can be further improved by making use of different notations in business process model and adjusting or modifying the method if necessary in order to widen its applicability.

Given that the e-Cosmic method underestimates the real software size, further research can be conducted on the e-Cosmic method to see if this underestimation is systematic. Then, it may be possible to understand if the underestimation takes place by a constant ratio. Once this constant has been identified, e-Cosmic results can be increased by making use of the identified constant to make the results converge to real software sizes.

REFERENCES

Abran, A. (1999). *COSMIC FFP 2.0: An Implementation of COSMIC Functional Size Measurement Concepts*, FESMA'99, October7, Amsterdam, Holland.

Abran, A., Desharnais, J.-M., Oligny, S., St-Pierre, D. & Symons, C. (2003). *COSMIC FFP - Measurement Manual – the COSMIC Implementation Guide for ISO/IEC 19761: January*.

Abran, A., Desharnais, J.M., Lesterhuis, A., Londeix, B., Meli, R., Morris, P., Oligny, S., O'Neil, M., Rollo, T., Rule, G., Santillo, L., Symons, C. & Toivonen, H. (2007). *The COSMIC Functional Size Measurement Method Manual*, version 3.0. GELOG.

Abrahao, S., and Insfran, E., (2008). *A Metamodeling Approach to Estimate Software Size from Requirements Specifications*, 34 th Euromicro Conference Software Engineering and Advanced Applications,SEAA.2008.53.

Albrecht, A. (1979). *Measuring Application Development Productivity*. Proceedings of the Joint SHARE/GUIDE/IBM Application Development Symposium, (pp. 83-92).

Azzouz, S. & Abran, A. (2004). *A proposed measurement role in the Rational Unified Process (RUP) and its implementation with ISO 19761: COSMIC FFP*. Proceedings of the Software Measurement European Forum 2004, Rome , Italy.

Boehm, B. (1981). *Software Engineering Economics*. New Jersey: Prentice Hall.

Condori-Fernández, N., Abrahão, S. & Pastor, O. (2007). On the Estimation of Software Functional Size from Requirements Specifications. *Journal of Computer Science and Technology, 22(3)*, 358–370.

Conte, M. Iorio, T. Meli, R. & Santillo, L. (2004). *E&Q: An Early & Quick Approach to Functional Size Measurement Methods*, in Software Measurement European Forum (SMEF), Rome, Italy.

Davis, R. & Brabander, E. (2007). *ARIS Design Platform Getting Started with BPM*, London: Springer- Verlag.

Demirörs, O. & Gencel, Ç. (2004). *A Comparison of Size Estimation Techniques Applied Early in the Life Cycle*, European Software Process Improvement Conference (EurSPI 2004), Lecture Notes in Computer Science (LNCS), pp.184-194.

Diab, H., Koukane, F., Frappier, M. & St-Denis, R. (2005). µcROSE: Automated Measurement of COSMIC-FFP for Rational Rose Real Time*. Information and Software Technology, 47(3)*,151–166.

Fenton, N. E. & Pfleeger, S. L. (1997). *Software Metrics: A Rigorous and Practical Approach, Second Edition*. Boston, MA :International Thomson Computer Press.

Fetcke, T. (1999). *A generalized structure for function point analysis*. Proceedings of International Workshop on Software Measurement (IWSM'99), Mont-Tremblant, Canada, 1-11.

Fetcke, T., Abran, A. & Nguyen, T. (1997). *Mapping the OO - Jacobson, Approach into Function Point Analysis, Software*. Proceedings of Technology of Object-Oriented Languages and Systems 23. Santa Barbara, CA, USA, 192-202.

Galorath D. D. & Evans M. W. (2006). *Software sizing, estimation, and risk management : when performance is measured performance improves*. USA : Auerbach Publications.

Glass, R.L. (2003). *Facts and Fallacies of Software Engineering*. Boston, MA : Addison Wesley.

Hericko, M., Rozman, I. & Zivkovic, A. (2006). A formal representation of functional size measurement methods. *The journal of Systems and Software, 79(9),* 1341-1358.

Hlupic, V. & Robinson, S., (1998). B*usiness Process Modeling and Analysis Using Discrete-Event Simulation*, Proceedings of the Winter Simulation Conference, 2, (pp. 1363-1369).

ISO/IEC (1998). 14143R1: Information Technology – Software Measurement Functional Size Measurement Part 1: Definition of Concepts.

ISO/IEC (2002). 20968:2002: Software Engineering – Mark II Function Point Analysis-Counting Practices Manual.

ISO/IEC (2003a). 19761:2003: Software Engineering COSMIC FFP: A Functional Size Measurement Method.

ISO/IEC (2003b). 20926:2003: Software Engineering – IFPUG 4.1 Unadjusted FSM Method-Counting Practices Manual.

Jones, C. (2007). *Estimating Software Costs : Bringing Realism to Estimating*. New York : McGraw-Hill Companies.

Jones, C. (2008). *Applied Software Measurement : Global Analysis of Productivity and Quality.* New York : McGraw-Hill.

Lamma, E., Mello, P. & Riguzzi, F. (2004). A System for Measuring Function Points. *Computer journal, 47(3) ,* 358-372.

Levesque, G., Bevo, V. & Cao, D.T. (2008). *Estimating software size with UML models*. Proceedings of the 2008 C3S2E Conference, Montreal, 81–87.

Marin, B., Giachetti, G. & Pastor, O. (2008). *Measurement of Functional Size in Conceptual Models: A Survey of Measurement Procedures Based on COSMIC* , IWSM / MetriKon / Mensura 2008 , LNCS, vol 5338/2008 , (pp. 170-183).

Meli, R. (1997). *Early and Extended Function Point: A New Method for Function Points Estimation*, IFPUG-Fall Conference, September 15-19, Scottsdale, Arizona, USA.

Meli, R. & Santillo L. (1999). *Function Point Estimation Methods: A Comparative overview*, FESMA 98-The European Software Measurement Conference, October 6-8. , Amsterdam, Holland.

Meli, R., Abran, A., Ho, V.T. & Oligny, S. (2000). *On the Applicability of COSMIC FFP for Measuring Software Throughout Its Life Cycle*, Proceedings of the ESCOM-SCOPE 2000, April, Munich, Germany, Shaker Publ., (pp. 289-297).

Poels, G. (2003). *Functional Size Measurement of Multi-Layer Object-Oriented Conceptual Models.* Proceedings of the 9th International Object-Oriented Information Systems Conference, Geneva, (pp. 334–345).

Poels, G. (2002). *A Functional Size Measurement Method for Event-Based Object-Oriented Enterprise Models*. Proceedings of the 4th International Conference on Enterprise Information Systems –ICEIS, Ciudad Real, (pp. 667–675).

Rask, R. (1991). *Algorithms for Counting Unadjusted Function points from Dataflow Diagrams*, Research report, University of Joensuu, Finland.

Santillo, L. & Meli, R. (1998). *Early Function Points: some practical experiences of use*, ESCOM-ENCRESS 98, May 18, Roma, Italy.

Scheer, A.W. (2001). *ARIS Methods*, Berlin: Springer.

Symons, C. (2001). *Come Back Function Point Analysis (Modernized) – All is Forgiven!),* Proceedings of the 4th European Conference on Software Measurement and ICT Control, FESMA-DASMA 2001, Germany,(pp. 413-426).

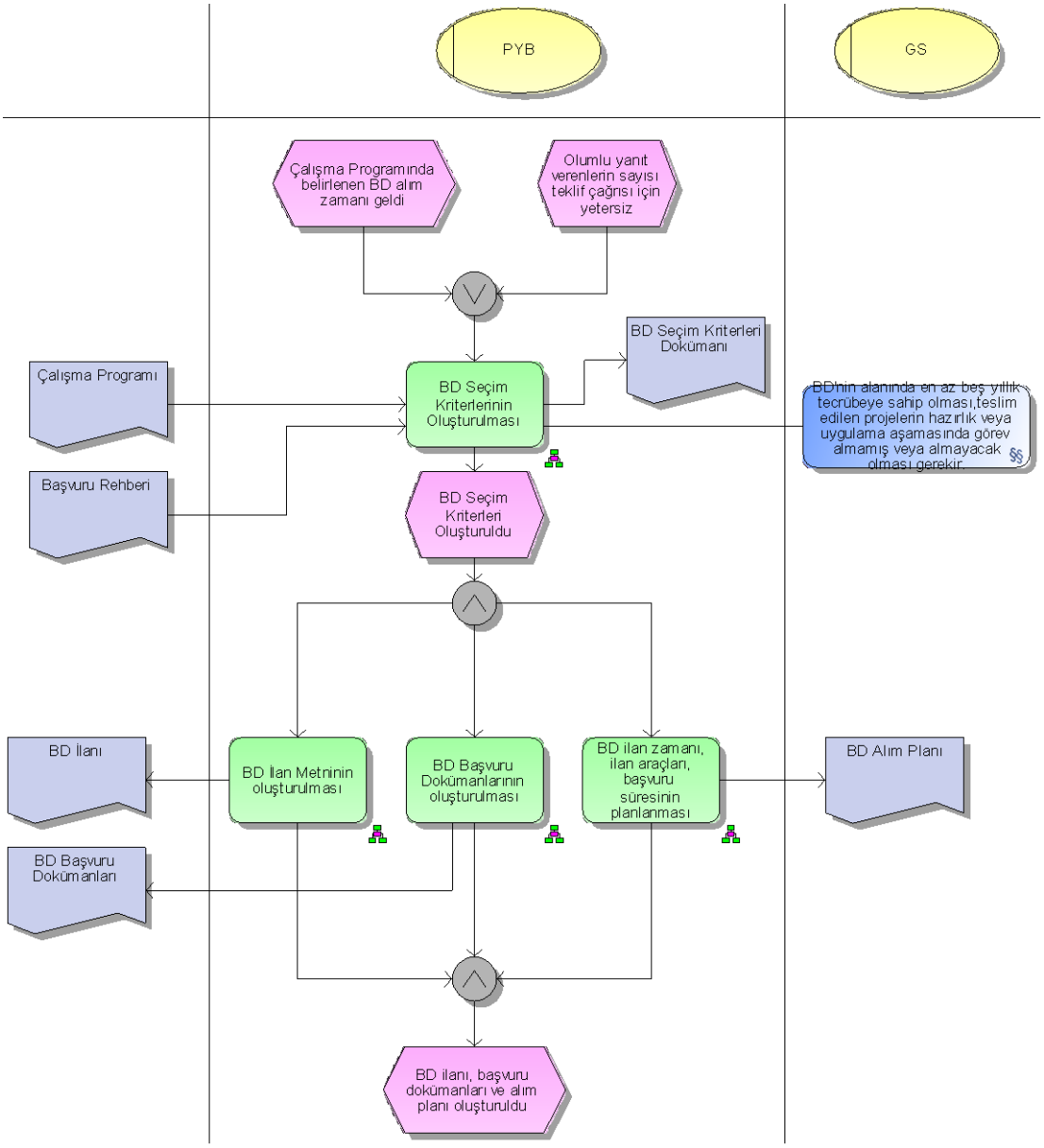
Uemura, T., Kusumoto, S. & Inoue, K. (1999). *Function Point Measurement Tool for UML Design Specification.* Proceedings of the 5th International Software Metrics Symposium, IEEE Metrics, Florida, USA, (pp. 62-71).

van den Berg, K.G., Dekkers, T. & Oudshoorn, R. (2005). *Functional Size Measurement applied to UML-based user requirements.* Proceedings of the 2nd Software Measurement European Forum (SMEF2005), 16-18 March 2005, Rome, Italy.

Vogelezang, F. (2005). *Early estimating using COSMIC-FFP*, SMEF'05 Conference proceedings, Rome, 16-18 March.

APPENDICES

APPENDIX A: BUSINESS PROCESS MODEL



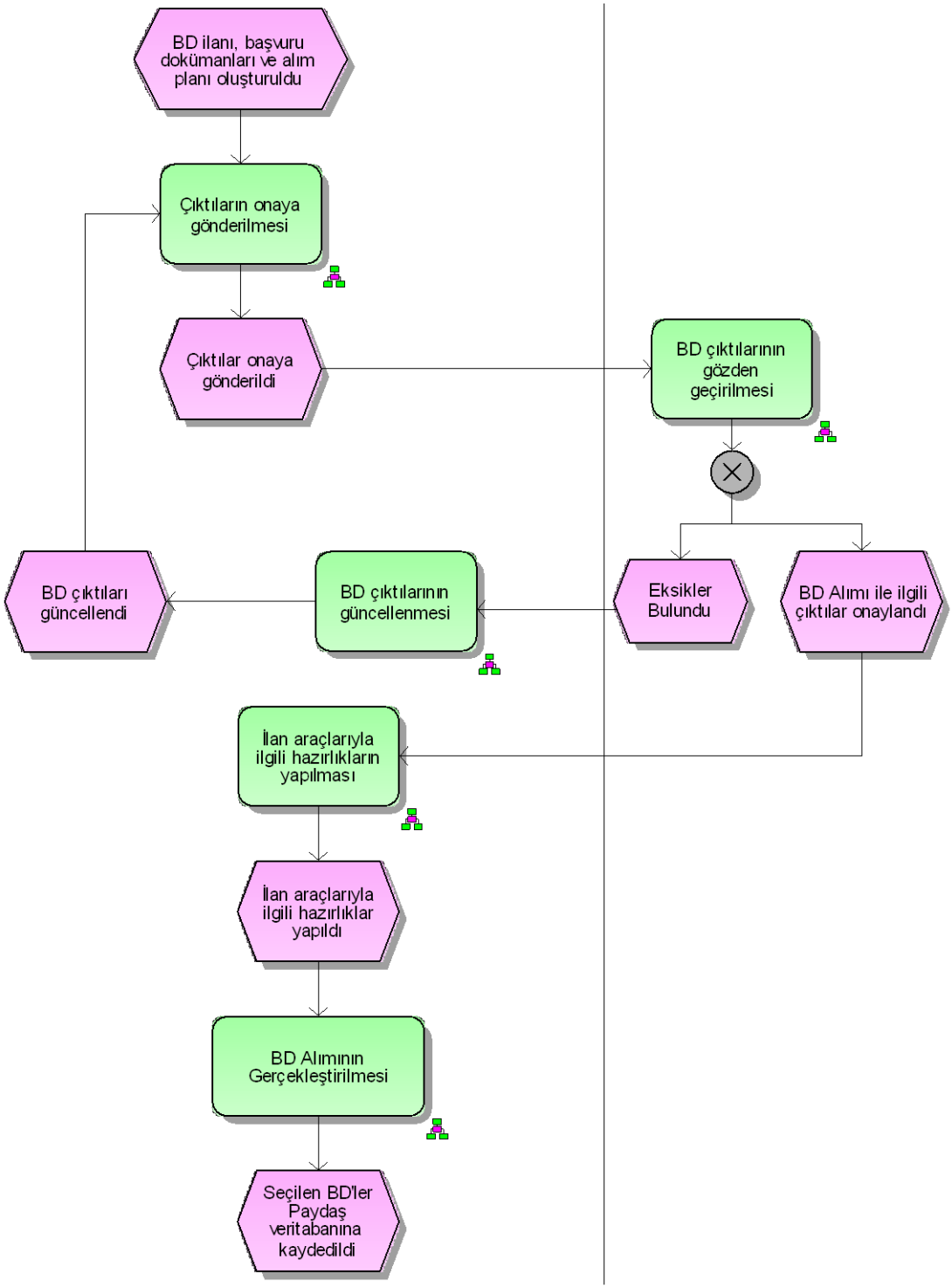


Figure 16 : Business Process Model instance

APPENDIX B: USE CASE MODEL AND COSMIC FSM

BAĞIMSIZ DEĞERLENDİRİCİLERİN BELİRLENMESİ SÜRECİ

Use Case	#1: BD seçim kriterlerinin oluşturulması
Primary Actor	PYB Uzmanı
Stakeholders	PYB Uzmanı
Minimal Guarantees	-
Success Guarantees	Seçim kriterleri oluşturuldu.
Preconditions	-
Triggers	Çalışma programında belirlenen BD alım zamanının gelmesi.
Main Scenario Success	1- PYB uzmanı başvuru rehberi görüntüleme isteğinde bulunur. 2- Sistem başvuru rehberi konularını listeler. 3- PYB uzmanı teklif çağrısı konusu bölümünü seçer. 4- Sistem teklif çağrısı konusu bölümünü görüntüler. 5- PYB uzmanı teklif çağrısı konusunu düzenler. 6- PYB uzmanı teklif çağrısını seçim kriterlerine ekler. 7- Sistem seçim kriterlerini kaydeder. 8- PYB uzmanı çalışma programından bağımsız değerlendiricilerin seçimi ve görevlendirilmesi bilgilerini görüntüleme isteğinde bulunur. 9- Sistem bağımsız değerlendiricilerin seçimi ve görevlendirilmesi bilgilerini görüntüler. 10- PYB uzmanı gerekli düzenlemeyi yapıp seçim kriterlerine ekler. 11- Sistem seçim kriterlerini kaydeder.
Extensions	6-11 a- PYB uzmanı seçim kriterlerini silme isteğinde bulunur. 1-PYB uzmanı seçim kriterlerini seçer. 2- PYB uzmanı seçilen seçim kriterlerini siler. 3- Sistem değişikleri kayıt eder. 6-11 b- PYB uzmanı seçim kriterlerini değiştirme isteğinde bulunur. 1- PYB uzmanı seçim kriterlerini değiştirir. 2- Sistem değişiklikleri kaydeder .

Use Case	#2: BD ilan metninin oluşturulması
Primary Actor	PYB Uzmanı
Stakeholders	PYB Uzmanı
Minimal Guarantees	-
Success Guarantees	İlan metni oluşturuldu.
Preconditions	Seçim kriterleri oluşturuldu.
Triggers	Seçim kriterleri oluşturuldu.
Main Scenario Success	1- PYB uzmanı sisteme girer 2- PYB uzmanı ilan ekleme isteğinde bulunur. 3- Sistem yeni bir ilan metni formu açar. 4- PYB uzmanı seçim kriterlerini görüntüleme isteğinde bulunur. 5- Sistem seçim kriterlerini listeler. 6- PYB uzmanı seçim kriterlerini seçerek ilan metnine ekler. 7- PYB uzmanı ilan metni formunu düzenler. 8- Sistem ilan metni formu bilgilerini kaydeder.
Extensions	2-a PYB uzmanı ilan metni formunu silme isteğinde bulunur. 1- Sistem ilan metni formlarını listeler. 2- PYB uzmanı ilan metnini seçerek silme isteğinde bulunur. 3- Sistem ilan metnini siler. 2-b- PYB uzmanı ilan metni formunu değiştirme isteğinde bulunur. 1- Sistem ilan metni formlarını listeler. 3- PYB uzmanı ilan metni seçer. 4- Sistem ilan formunu görüntüler. 5- PYB uzmanı ilan formunu değiştirir. 6- Sistem ilan metni formunu kaydedip 2'e döner.

Use Case	#3: BD başvuru dökümanlarının oluşturulması
Primary Actor	PYB Uzmanı
Stakeholders	PYB Uzmanı
Minimal Guarantees	-
Success Guarantees	BD başvuru dökümanı oluşturuldu.
Preconditions	Seçim kriterleri oluşturuldu.
Triggers	Seçim kriterleri oluşturuldu.
Main Scenario Success	1- PYB uzmanı sisteme girer. 2- PYB uzmanı yeni BD başvuru dökümanını oluşturma isteğinde bulunur. 3- Sistem BD şablonunu görüntüler. 4- PYB uzmanı standart BD başvuru dökümanı oluşturacak

	şekilde şablonu doldurur. 5- Sistem şablonu kaydeder. 6- Sistem BD başvuru dökümanını görüntüler.
Extensions	3 a- PYB BD başvuru dökümanını silme isteğinde bulunur. 1- Sistem BD başvuru dökümanlarını listeler. 2- PYB uzmanı silmek istediği dökümanı seçer 3- Sistem BD başvuru dökümanını siler ve 2'ye döner.. 3 b- PYB uzmanı BD başvuru dökümanını değiştirme isteğinde bulunur. 1- Sistem BD başvuru dökümanlarını listeler. 2- PYB uzmanı BD başvuru dökümanını seçer. 3- Sistem BD başvuru dökümanını görüntüler. 4- PYB uzmanı BD başvuru dökümanını değiştirir. 5- Sistem BD başvuru dökümanını kaydedip 2'e döner.

Use Case	#4: BD ilan zamanı, ilan araçları, başvuru süresinin planlaması
Primary Actor	PYB Uzmanı
Stakeholders	PYB Uzmanı
Minimal Guarantees	-
Success Guarantees	BD alım planı oluşturuldu.
Preconditions	Seçim kriterleri oluşturuldu.
Triggers	Seçim kriterleri oluşturuldu.
Main Scenario Success	1- PYB uzmanı sisteme girer 2- PYB uzmanı yeni BD alım planını oluşturma isteğinde bulunur. 3- Sistem alım planı şablonunu görüntüler. 4- PYB uzmanı ilan zamanı, ilan araçları ve başvuru süresi bilgilerini girer. 5- Sistem alım planını kaydeder.
Extensions	3 a- PYB BD başvuru dökümanını silme isteğinde bulunur. 1- Sistem BD alım planlarını listeler. 2- PYB uzmanı BD alım planı silme isteğinde bulunur. 3- Sistem BD alım planını siler ve 2'ye döner.. 3 b- PYB uzmanı BD alım planı değiştirme isteğinde bulunur. 1- Sistem BD alım planlarını listeler. 2- PYB uzmanı BD alım planını seçer. 3- Sistem BD alım planını görüntüler. 4- PYB uzmanı BD alım planını değiştirir. 5- Sistem BD alım planını kaydedip 2'e döner.

Use Case	#5: Çıktıların onaya gönderilmesi
Primary Actor	PYB Uzmanı
Stakeholders	GS, PYB Uzmanı
Minimal Guarantees	-
Success Guarantees	Çıktılar onaya gönderildi.
Preconditions	BD ilanı, başvuru dökümanları ve alım planı oluşturuldu.
Triggers	BD ilanı, başvuru dökümanları ve alım planı oluşturuldu.
Main Scenario Success	1- PYB uzmanı BD ilanı, başvuru dökümanları ve alım planının onaya gönderilmesi isteğinde bulunur. 2- Sistem BD ilanı, başvuru dökümanları ve alım planılarını listeler. 3- PYB uzmanı ilgili BD ilanı, başvuru dökümanları ve alım planlarını seçer ve onay isteğinde bulunur. 4- Sistem alım planı şablonunu seçilen BD ilanı, başvuru dökümanları ve alım planını onay için sorumlu GS'ye gönderir . 5- Sistem gönderme zamanı ve onay isteği bilgilerini kaydeder.
Extensions	-

Use Case	#6: BD çıktılarının gözden geçirilmesi.
Primary Actor	GS uzmanı
Stakeholders	GS uzmanı
Minimal Guarantees	-
Success Guarantees	Çıktılar onaylandı.
Preconditions	Çıktılar onaya gönderildi.
Triggers	Çıktılar onaya gönderildi.
Main Scenario Success	1- GS uzmanı çıktıların listelenmesi isteğinde bulunur. 2- Sistem onay isteği, BD başvuru dökümanları, BD alım planı, BD ilan metni, BD seçim kriterlerini listeler. 3- GS uzmanı ilgili dökümanı seçerek görüntüleme isteğinde bulunur. 4- Sistem dökümanı görüntüler. 5- GS dökümanı onaylar. 6- Sistem onay bilgisini kaydeder.
Extensions	5 a- GS uzmanı dökümandaki eksiklikleri belirler. 1- İlgili dökümandaki eksikleri yorum alanına yazar. 2- Sistem yorumları kaydeder 1 'e döner.

Use Case	#7: BD çıktılarının güncellenmesi.
Primary Actor	PYB uzmanı
Stakeholders	PYB uzmanı
Minimal Guarantees	-
Success Guarantees	BD çıktıları güncellendi
Preconditions	Eksiklikler bulundu.
Triggers	Eksiklikler bulundu.
Main Scenario Success	1- PYB uzmanı çıktıların güncellenmesi isteğinde bulunur. 2- Sistem onay isteği, BD başvuru dökümanları, BD alım planı, BD ilan metni, BD seçim kriterlerini ve onay durumunu listeler. 3- PYB uzmanı ilgili dökümanı seçerek görüntüleme isteğinde bulunur. 4- Sistem dökümanı ve GS uzmanının yorumlarını görüntüler. 5- PYB uzmanı dökümanları günceller. 6- Sistem dökümanları kaydedip, onaya gönderir.
Extensions	-

Use Case	#8: İlan araçlarıyla ilgili hazırlıkların yapılması
Primary Actor	PYB uzmanı
Stakeholders	PYB uzmanı
Minimal Guarantees	-
Success Guarantees	Ilan araçları ile ilgili hazırlıklar yapıldı.
Preconditions	BD alımı ile ilgili çıktılar onaylandı.
Triggers	BD alımı ile ilgili çıktılar onaylandı.
Main Scenario Success	1- PYB uzmanı ilan araçları bilgisini görüntüleme isteğinde bulunur. 2- Sistem ilan araçlarını ve zaman bilgilerini görüntüler. 3- PYB uzmanı ilgili tarih ve ilan araç bilgilerini düzenler. 4- Sistem ilan araçları bilgilerini kaydeder.
Extensions	-

Use Case	#9: BD alım ilanının internet sitesinde yayınlanması.
Primary Actor	PYB uzmanı
Stakeholders	PYB uzmanı
Minimal Guarantees	-

Success Guarantees	BD alım ilanı internet sitesinde yayınlandı.
Preconditions	BD alım planında belirlenen ilan zamanı ve ilan araçları hazırlıkları yapıldı.
Triggers	BD alım planında belirlenen ilan zamanı ve ilan araçları hazırlıkları yapıldı.
Main Scenario Success	1- PYB uzmanı BD alım ilanının yayınlanması isteğinde bulunur. 2- Sistem BD alım planı, BD ilanı ve BD başvuru dökümanlarını listeler. 3- PYB uzmanı BD ilanını ve başvuru dökümanlarını seçip yayınlanması isteğinde bulunur. 4- Sistem ajans internet sitesinde yayınlanmak üzere ilanı kaydeder.
Extensions	-

Use Case	#10: BD Başvurularının alınması.
Primary Actor	BD adayı
Stakeholders	BD adayı, PYB uzmanı
Minimal Guarantees	-
Success Guarantees	BD başvuruları alındı.
Preconditions	BD alım ilanı internet sitesinde yayınlandı.
Triggers	BD alım ilanı internet sitesinde yayınlandı.
Main Scenario Success	1- BD adayı BD başvuru formu doldurma isteğinde bulunur. 2- Sistem BD başvuru formunu görüntüler. 3- BD adayı başvuru formunu doldurur. 4- Sistem BD başvuru formunu kaydeder.
Extensions	-

Use Case	#11: Seçim komisyonunun belirlenmesi.
Primary Actor	GS
Stakeholders	GS
Minimal Guarantees	-
Success Guarantees	seçim komisyonu belirlendi
Preconditions	BD alım ilanı internet sitesinde yayınlandı.
Triggers	BD alım ilanı internet sitesinde yayınlandı.
Main Scenario Success	1- GS uzmanı seçim komisyonunu belirleme isteğinde

	bulunur. 2- Sistem ajans uzmanlarını, birim ve pozisyon bilgileri ile listeler. 3- GS uzmanı 3 tane seçim kurulu komisyonu üyesini listeden seçer. 4- Sistem BD seçim komisyonu üye bilgilerini kaydeder.
Extensions	4-a . üye sayısının 3 den az seçilmesi ve seçim kurallarına uyulmaması durumu 1. Sistem hata mesajı verir. 2. Sistem 2. Adıma geri döner

Use Case	#12: BD başvurularının değerlendirilmesi.
Primary Actor	BD seçim komisyonu
Stakeholders	BD seçim komisyonu
Minimal Guarantees	-
Success Guarantees	BD başvuruları değerlendirildi.
Preconditions	BD başvuruları alındı ve seçim komisyonu belirlendi.
Triggers	BD başvuruları alındı ve seçim komisyonu belirlendi.
Main Scenario Success	1- BD seçim komisyonu üyesi BD başvuru değerlendirme isteğinde bulunur. 2- Sistem BD adaylarını ve seçim kriterlerini listeler. 3- Sistem aday değerlendirme bilgilerini kaydedip, sonraki adaya geçer. 3-4 bütün adaylar bitene kadar devam eder. 4- Sistem seçilen adayları listeler.
Extensions	3-a. BD seçim komisyonu üyesi aday değerlendirme bilgisi değiştirme isteğinde bulunur. 1. Sistem aday bilgisini görüntüler. 2. BD seçim komisyonu üyesi değişiklikleri yapar. 3. Sistem değişiklikleri kaydeder.

Use Case	#13: Seçilen BD'lerin onaylanması ve paydaş veri tabanına kayıt edilmesi
Primary Actor	GS,PYB
Stakeholders	GS,PYB
Minimal Guarantees	-
Success Guarantees	Seçilen BD'ler onaylandı ve paydaş veri tabanına kayıt edildi
Preconditions	BD başvuruları değerlendirildi.
Triggers	BD başvuruları değerlendirildi.
Main Scenario Success	1- GS uzmanı seçilen adayları onaylama isteğinde bulunur . 2- Sistem seçilen BD listesini görüntüler. 3- GS uzmanı BD listesini onaylar. 4- Sistem onayı kaydeder. 5- PYB uzmanı onaylanan seçilen BD'leri paydas veri tabanına aktarma isteğinde bulunur. 6- System BD'leri paydaş veri tabanına kaydeder.
Extensions	3-a. GS uzmanı seçilen BD lerin yeniden değelendirmesi isteğinde bulunur. 1. Sistem BD seçim komisyonu üyelerini uyarır 1'e döner

FP#	FP name	Data Mov.	OOI/Data Group
1	BD seçim kriterlerinin oluşturulması	E	Bd seçim kriterleri oluşturma isteği
		E	Başvuru rehberi
		R	Başvuru rehberi
		X	Başvuru rehberi
		E	Teklif çağrısı
		R	Teklif çağrısı
		X	Teklif çağrısı
		R	Çalışma programı
		W	Bd seçim kriterleri
		X	Hata bilgisi
2	BD seçim kriterlerinin listelenmesi	E	Seçim kriterleri parametresi
		R	Seçim kriterleri
		X	Seçim kriterleri

FP#	FP name	Data Mov.	OOI/Data Group
3	BD seçim kriterlerinin silinmesi	E	Seçim kriterleri parametresi
		W	Seçim kriterleri
4	BD seçim kriterlerinin güncellenmesi	E	Seçim kriterleri parametresi
		W	Seçim kriterleri
5	BD ilan metninin oluşturulması	E	Bd ilan metni isteği
		R	Seçim kriterleri bilgisi
		X	Seçim kriterleri bilgisi
		E	Seçim kriterleri bilgisi
		E	Bd ilan metni bilgisi
		W	Bd ilan metni bilgisi
		X	Bd ilan metni bilgisi
6	BD ilan metninin listelenmesi	E	Bd ilan metni listeleme isteği
		R	Bd ilan metinleri bilgisi
		X	Bd ilan metinleri bilgisi
7	BD ilan metninin silinmesi	E	Ilan metni parametresi
		W	Ilan metni bilgisi
8	BD ilan metninin güncellenmesi	E	Ilan metni parametresi
		W	Ilan metni bilgisi
9	BD Başvuru dökümanlarının oluşturulması	E	Bd başvuru dökümanları oluşturma isteği
		R	Bd başvuru döküman alanları
		X	Bd başvuru döküman alanları
		E	Başvuru döküman alan bilgisi
		W	Bd başvuru döküman alan bilgisi

FP#	FP name	Data Mov.	OOI/Data Group
10	BD başvuru dökman alanlarının listelenmesi	E	Bd başvuru döküman alan bilgisi
		R	Bd başvuru döküman alan bilgisi
		X	Bd başvuru döküman alan bilgisi
11	BD başvuru döküman alanlarının silinmesi	E	Bd başvuru döküman alan bilgisi
		W	Bd başvuru döküman alan bilgisi
12	BD başvuru döküman alanlarının güncellenmesi	E	Bd başvuru döküman alan bilgisi
		W	Bd başvuru döküman alan bilgisi
13	BD ilan zamanı, ilan araçları, başvuru süresinin planlanması	E	Alım planı bilgisi
		W	Alım planı bilgisi
14	BD alım planının listelenmesi	E	Bd alım planı
		R	Bd alım planı
		X	Bd alım planı
15	BD alım planının silinmesi	E	Alım planı parametresi
		W	Alım planı bilgisi
16	BD alım planının güncellenmesi	E	Alım planı parametresi
		W	Alım planı bilgisi
17	Çıktıların onaya gönderilmesi	E	Onay isteği
		R	Çıktı dökümanları bilgisi
		X	Çıktı dökümanları bilgisi
		E	Çıktı dökümanları bilgisi-onay
		W	Çıktı dökümanları bilgisi-onay

FP#	FP name	Data Mov.	OOI/Data Group
18	BD çıktılarının gözden geçirilmesi	E	Çıktı bilgileri görüntüleme isteği
		R	Çıktı bilgileri (başvuru dök., alım planı, ilan metni, seçim kriterleri)
		X	Çıktı bilgileri (başvuru dök., alım planı, ilan metni, seçim kriterleri)
		R	Onay bilgisi
		R	Bd başvuru dökümanları
		R	Bd alım planı
		R	Bd ilan metni
		R	Bd seçim kriterleri
		X	Onay bilgisi
		X	Bd başvuru dökümanları
		X	Bd alım planı
		X	Bd ilan metni
		X	Bd seçim kriterleri
		W	Onay bilgisi
19	BD çıktılarının güncellenmesi	E	Çıktı bilgileri görüntüleme isteği
		R	Onay durum bilgisi
		R	Bd başvuru dökümanları
		R	Bd alım planı
		R	Bd ilan metni
		R	Seçim kriterleri
		X	Onay durum bilgisi
		X	Bd başvuru dökümanları
		X	Bd alım planı
		X	Bd ilan metni
		X	Seçim kriterleri
		E	Bd başvuru dökümanları
		E	Bd alım planı
		E	Bd ilan metni
		E	Bd seçim kriterleri
		W	Bd başvuru dökümanları

FP#	FP name	Data Mov.	OOI/Data Group
		W	Bd alım planı
		W	Bd ilan metni
		W	Bd seçim kriterleri
20	Ilan araçlarıyla ilgili hazırlık yapılması	E	Ilan araçları bilgisi isteği
		R	Bd alım planı bilgisi
		X	Ilan araçları bilgisi
21	BD alım ilanının internet sitesinde yayınlanması	E	İnternet ilanı isteği
		R	Internet ilanı bilgisi(başvuru dökümanları, alım planı, ilan metni)
		X	Internet ilanı bilgisi(başvuru dökümanları, alım planı, ilan metni)
		R	Bd başvuru dökümanları
		R	Bd alım planı
		R	Bd ilan metni
		R	Bd seçim kriterleri
		X	Bd başvuru dökümanları
		X	Bd alım planı
		X	Bd ilan metni
		X	Bd seçim kriterleri
22	Başvuruların alınması	E	Başvuru alınması isteği
		R	Başvuru dökümanları
		E	Başvuru aday bilgisi
		W	Başvuru aday bilgisi
23	Seçim komisyonun belirlenmesi	E	Seçim komisyonu parametre bilgisi
		R	Seçim komisyonu üyeleri
		X	Seçim komisyonu üyeleri
		E	Seçim komisyonu bilgisi
		W	Seçim komisyonu bilgisi

FP#	FP name	Data Mov.	OOI/Data Group
24	BD başvuru değerlendirme	E	Değerlendirme isteği
		R	Bd seçim kriterleri
		X	Bd seçim kriterleri
		R	Bd başvuru bilgisi
		X	Bd başvuru bilgisi
		E	Onay isteği
		W	Onay isteği
		E	Kabul listesi
		W	Kabul listesi
		E	Seçim komisyonu raporu
		W	Seçim komisyonu raporu
		E	Bd
		W	Bd
25	Seçilen BD lerin onaylanması ve kayıt	E	Onaylama isteği
		R	Onay bilgisi
		X	Onay bilgisi
		R	Kabul listesi onay bilgisi
		X	Kabul listesi onay bilgisi
		E	Kabul listesi onay bilgisi
		W	Kabul listesi onay bilgisi
		E	Onay durumu
		W	Onay durumu
		E	Bd
		W	Bd
	Toplam CFP	137	

ÖDEMELERİN GERÇEKLEŞTİRİLMESİ SÜRECİ

Use Case	#1: Ön ödeme, ara ödeme ve nihai ödeme oranlarının belirlenmesi
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı
Minimal Guarantees	-
Success Guarantees	Ödeme oranları belirlendi
Preconditions	-
Triggers	Sözleşme imzalama dönemi tamamlandı
Main Scenario Success	1- İDB uzmanı sisteme girer 2- İDB uzmanı ödeme gerçekleştirme isteğinde bulunur 3- Sistem sözleşmesi tamamlanmış mevcut proje listesini sunar 4- İDB uzmanı belirlenen projeyi seçer 5- Sistem proje bilgilerini görüntüler 6- İDB uzmanı ödeme kontrol listesi, proje uygun maliyetleri, ön ödeme risk puanı, sözleşme dosyasını görüntüleme isteğinde bulunur 7- Sistem istenilen bilgileri görüntüler 8- İDB uzmanı ön ödeme, ara ödeme ve nihai ödeme miktarını belirler ve ödeme planını oluşturur. 9- Sistem ön ödeme miktarını toplam miktar ile karşılaştırıp, kayıt eder 10- İDB uzmanı onay isteği oluşturma isteğinde bulunur. 11- Sistem onay isteğini kayıt eder.
Extensions	9a sistem hata verir, gerekli koşulların sağlanmadığına dair 1. İDB uzmanı hata mesajını görüntüler 2. Sistem ön ödeme miktarının alt ve üst sınırını gösterir 3. İDB uzmanı yeni ön ödeme miktarını sisteme girer 1a sistem hata verir 9a.1. den devam eder 4. MSS da 9 dan devam eder 8a İDB uzmanı ön ödeme miktarını değiştirmek ister 1. Sistem değişiklikleri kayıt eder. 3 den devam eder

	8b İDB uzmanı ön ödeme miktarını silmek ister 1. Sistem ön ödeme miktarını siler 3 den devam eder

Use Case	#2: Ödeme oranlarının onaylanması
Primary Actor	GS uzmanı
Stakeholders	GS uzmanı
Minimal Guarantees	-
Success Guarantees	ön ödeme oranı onaylandı, azaltıldı, artırıldı, iptal edildi.
Preconditions	ödeme oranları belirlendi
Triggers	ödeme oranları belirlendi
Main Scenario Success	1- GS uzmanı sisteme girer 2- GS uzmanı onay isteklerini görüntüleme isteğinde bulunur 3- Sistem onay isteklerini listeler 4- GS uzmanı onay isteği detaylarını görüntüleme isteğinde bulunur, 5- Sistem onay isteği detaylarını görüntüler 6- GS uzmanı ön ödeme miktarı ve ödeme planını görüntüleme isteğinde bulunur 7- Sistem ön ödeme miktarını ve ödeme planını sunar 8- GS uzmanı onay durumunu yaratır 9- Sistem kayıt eder.
Extensions	8a GS uzmanı ön ödeme miktarının azaltılmasını ister 1. Sistem uyarı verir. 2. GS uzmanı ödeme planını görüntüleme isteğinde bulunur. 3. Sistem ödeme planını görüntüler 4. GS uzmanı ön ödeme oranı değişikliği bilgisini yaratır. 5. Sistem değişiklik bilgisini ödeme planına yansıtarak yararlanıcıya bilgi verir 8b GS uzmanı ön ödeme miktarının artırılmasını ister. 1. 8a. 1. den devam eder 8c GS uzmanı ön ödeme miktarının iptalini ister. 1. 8a.1. den devam eder 8d GS onay durumunu değiştirmek ister

	1. Sistem değişiklikleri kayıt edip 3 ‘ döner 8e GS onay durumunu silmek ister 1. Sistem değişiklikleri kayıt eder 3’e döner

Use Case	#3: Başvuru sahibinden ön ödeme talebi alınması
Primary Actor	Yararlanıcı
Stakeholders	Yararlanıcı
Minimal Guarantees	-
Success Guarantees	Başvuru sahibinden ön ödeme talebi alındı.
Preconditions	
Triggers	
Main Scenario Success	1- Yararlanıcı sisteme girer. 2- Yararlanıcı ön ödeme talebi oluşturma isteğinde bulunur 3- Sistem ödeme talebi şartlarını görüntüler 4- Yararlanıcı ödeme şartlarına uygunluğunu kabul eder. 5- Sistem ön ödeme talebini kayıt eder. 6- Yararlanıcı teminat bilgilerini girer 7- Sistem teminat bilgilerini kontrol eder. 8- Sistem teminat bilgilerini kayıt eder.
Extensions	8a . sistem hata mesajı verir, teminatın uygun olmadığına dair 1. Yararlanıcı teminat bilgilerini tekrar girer 7 den devam eder 6a yararlanıcı teminat bilgilerini değiştirmek ister 1. Sistem değişiklikleri kayıt eder 3 ‘e döner 6b yararlanıcı teminat bilgilerini silmek ister 1. Sistem değişiklikleri kayıt eder 3 ‘e döner

Use Case	#4: Ön ödeme miktarının hesap numarasına aktarılması.
Primary Actor	İDB Uzmanı
Stakeholders	İDB Uzmanı

Minimal Guarantees	-
Success Guarantees	Ön ödeme gerçekleştirildi
Preconditions	Teminat var
Triggers	
Main Scenario Success	1- İDB Uzmanı Proje detaylarını seçer 2- Sistem proje detaylarını görüntüler. 3- İDB Uzmanı belirtilen hesap numarasına ön ödeme talimatı oluşturur. 4- Sistem muhasebe departmanına ödeme talimatını aktırır.
Extensions	

Use Case	#5: hakediş esasına göre ön ödeme harcamalarını değerlendirme
Primary Actor	İDB Uzmanı
Stakeholders	İDB Uzmanı
Minimal Guarantees	-
Success Guarantees	Ön ödeme miktarı kullanılmamış, ön ödeme miktarı kullanılmamış
Preconditions	Ön ödeme gerçekleştirildi
Triggers	
Main Scenario Success	1- İDB Uzmanı sisteme girer 2- İDB Uzmanı proje detaylarını görüntüleme isteğinde bulunur 3- Sistem proje detaylarını görüntüler. 4- İDB Uzmanı eş finansman oranı, ön ödeme miktarı, ödeme talebi, ödeme planı, ara dönem hakediş miktarını ve ödeme kontrol listesini görüntüleme isteğinde bulunur 5- Sistem eş finansman oranı, ön ödeme miktarı, ödeme talebi, ödeme planı, ara dönem hakediş miktarını ve ödeme kontrol listesini görüntüler 6- İDB Uzmanı ön ödemenin kapatılması isteğinde bulunur. 7- Sistem ödeme planını görüntüler 8- İDB Uzmanı ödeme planının değiştirir 9- Sistem değişikliği kayıt eder.

Extensions	6a İDB Uzmanı ön ödemede kalan harcanmamış miktarı tespit eder 1. İDB Uzmanı ön ödeme miktarı ve ara dönem hakediş miktarını görüntüleme isteğinde bulunur. 2. Sistem ön ödeme miktarını ve ara dönem hakediş miktarını görüntüler 3. İDB Uzmanı hakediş gerçekleşmemiş ödeme miktarını oluşturur 3a- hakediş gerçekleşmemiş ödeme miktarını değiştirmek ister 1. Sistem değişiklikleri kayıt eder 3b -hakediş gerçekleşmemiş ödeme miktarını silmek ister 1. Sistem değişiklikleri kayıt eder 4. Sistem kayıt eder.

Use Case	#6: Ara ödeme miktarının oluşturulması
Primary Actor	İDB Uzmanı
Stakeholders	İDB Uzmanı
Minimal Guarantees	-
Success Guarantees	Ara ödeme miktarı belirlendi
Preconditions	Ön ödeme kapatıldı
Triggers	Ön ödeme kapatıldı
Main Scenario Success	1- İDB Uzmanı sisteme girer 2- İDB Uzmanı proje detaylarını görüntüleme isteğinde bulunur 3- Sistem proje detaylarını görüntüler 4- İDB Uzmanı eş finansman oranı, başvuru rehberi ve proje uygun maliyetleri görüntüleme isteğinde bulunur 5- Sistem eş finansman oranı, başvuru rehberi ve proje uygun maliyetleri görüntüler 6- İDB Uzmanı ödeme planı görüntüleme isteğinde bulunur 7- Sistem ödeme planını görüntüler 8- İDB Uzmanı ödeme planını değiştirir 9- Sistem değişikliği kayıt eder 10- İDB Uzmanı onay isteği oluşturur 11- Sistem onay isteğini kayıt eder

Extensions	10a -İDB uzmanı onay isteğini değiştirmek ister 1. Sistem değişiklikleri kayıt eder 10b- İDB uzmanı onay isteğini silmek ister

Use Case	#7: Ara dönem ödeme miktarının onaylanması ve hesaba transferi
Primary Actor	İDB Uzmanı, GS
Stakeholders	İDB Uzmanı, GS, yararlanıcı
Minimal Guarantees	-
Success Guarantees	Ara ödeme miktarı yararlanıcı hesabına aktarıldı.
Preconditions	Ara dönem ödeme miktarı tespit edildi
Triggers	
Main Scenario Success	1- GS sisteme girer 2- GS onay isteklerini görüntüleme isteğinde bulunur 3- Sistem onay isteklerini görüntüler 4- GS ara dönem ödeme miktarı ödeme planı ve onay isteği yazısını görüntüleme isteğinde bulunur 5- Sistem ara dönem ödeme miktarı ödeme planı ve onay isteği yazısını görüntüler. 6- GS onay Durumu oluşturur 7- Sistem onay durumunu kayıt eder 8- İDB uzmanı onay durumlarını görüntüleme isteğinde bulunur. 9- Sistem onay durumlarını görüntüler 10- İDB uzmanı onaylanmış miktarları sözleşme hesap numaralarına aktarır 11- Sistem para aktarma işlemini kayıt eder.
Extensions	---

Use Case	#8: Proje toplam uygun maliyetlerinin belirlenmesi
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı
Minimal Guarantees	-
Success Guarantees	Eş finansman gerçekleşme oranı saptandı
Preconditions	Ara veya nihai rapor incelenmesi tamamlandı
Triggers	Proje ara ödemesi gerçekleşti
Main Scenario Success	1- İDB uzmanı sisteme girer 2- İDB uzmanı proje görünteleme isteğinde bulunur 3- sistem projeyi görüntüler 4- İDB uzmanı ön ödeme miktarı, ara dönem ödeme miktarı, proje uygun maliyetleri, temel proje raporları, sözleşme dosyası , ödeme kontrol listesi görüntüleme isteğinde bulunur 5- Sistem ön ödeme miktarı, ara dönem ödeme miktarı, proje uygun maliyetleri, temel proje raporları, sözleşme dosyası , ödeme kontrol listesini görüntüler 6- İDB uzmanı proje nihai destek miktarını oluşturur 7- Sistem proje nihai destek miktarını kayıt eder.
Extensions	6a - İDB uzmanı nihai destek miktarını değiştirmek ister 1. Sistem değişiklikleri kayıt eder 6b- İDB uzmanı nihai destek miktarını silmek ister 1. Sistem değişiklikleri kayıt eder 7a sistem eş finansman gerçekleşme oranı hakkında hata verir 1. İDB uzmanı başvuru rehberini görüntüleme isteğinde bulunur 2. Sistem başvuru rehberini görüntüler 3. İDB uzmanı eş finansman gerçekleşme oranında proje nihai destek miktarını indirir 4. Sistem yeni proje destek miktarını kayıt eder.

Use Case	#9: Nihai ödemenin belirlenmesi
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı

Minimal Guarantees	-
Success Guarantees	Nihai ödeme miktarı belirlendi
Preconditions	Toplam ödeme miktarı belirlendi
Triggers	Toplam ödeme miktarı belirlendi
Main Scenario Success	1- İDB uzmanı sisteme girer 2- İDB uzmanı proje görüntüleme isteğinde bulunur 3- Sistem projeyi görüntüler 4- İDB uzmanı proje nihai destek miktarı, ön ödeme miktarı , ara dönem ödeme miktarı görüntüleme isteğinde bulunur 5- Sistem proje nihai destek miktarı, ön ödeme miktarı , ara dönem ödeme miktarını görüntüler 6- İDB uzmanı proje nihai ödeme miktarını oluşturur 7- Sistem proje nihai ödeme miktarını kayıt eder 8- İDB uzmanı ödeme planına proje nihai ödeme mikatarını girer 9- Sistem ödeme planını kayıt eder. 10- İDB uzmanı proje nihai ödeme miktarı için onay isteği oluşturur 11- Sistem onay isteğini kayıt eder.
Extensions	6a - İDB uzmanı nihai ödeme miktarını değiştirmek ister 2. Sistem değişiklikleri kayıt eder 6b- İDB uzmanı nihai ödeme miktarını silmek ister 2. Sistem değişiklikleri kayıt eder

Use Case	#2: Nihai ödeme miktarının onaylanması
Primary Actor	GS uzmanı
Stakeholders	GS uzmanı
Minimal Guarantees	-
Success Guarantees	Nihai ödeme gerçekleşti
Preconditions	Nihai ödeme miktarı belirlendi.
Triggers	
Main Scenario	1- GS sisteme girer 2- GS onay isteklerini görüntüleme isteğinde bulunur

Success	3- Sistem onay isteklerini görüntüler 4- GS proje nihai ödeme miktarı ve ödeme planı görüntüleme isteğinde bulunur 5- Sistem proje nihai ödeme miktarı ve ödeme planı görüntüler 6- GS proje onay durumunu oluşturur 7- Sistem proje onay durumunu kayıt eder
Extensions	6a – GS onay durumunu değiştirmek ister 1. Sistem değişiklikleri kayıt eder 6b- GS onay durumunu silmek ister 1. Sistem değişiklikleri kayıt eder

Use Case	#9: Nihai ödeme miktarının hesap numarasına aktarılması
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı
Minimal Guarantees	-
Success Guarantees	Proje nihai ödemesi gerçekleşti
Preconditions	Nihai ödeme miktarı onaylandı
Triggers	Toplam ödeme miktarı belirlendi
Main Scenario Success	1- İDB uzmanı sisteme girer 2- İDB uzmanı projeyi görüntüleme isteğinde bulunur 3- Sistem projeyi görüntüler 4- İDB uzmanı proje nihai ödeme miktarını görüntüleme isteğinde bulunur 5- Sistem nihai ödeme miktarını görüntüler 6- İDB uzmanı nihai ödeme miktarını hesap numarasına ödenmesi talimatı yaratır 7- Sistem hesap numarasına ödeme talimatını kayıt eder.
Extensions	6a- İDB uzmanı kalan eksi değerin ajans hesabına ödenmesi talimatını yaratır 1. Sistem geri ödeme talimatını kayıt eder 2. İDB uzmanı yararlanıcıya geri ödeme miktarını iletir 3. Sistem yararlanıcıya uyarı mesajı atar, yararlanıcıdan geri ödeme alınana kadar blok işlemi gerçekleşir.

FP#	FP name	Data Mov.	OOI/Data Group
1	Ödeme oranlarının belirlenmesi	E	Ödeme oranları
		E	Proje listesi
		R	Proje listesi
		X	Proje listesi
		E	Proje
		R	Proje
		X	Proje
		E	Ödeme kontrol listesi
		R	Ödeme kontrol listesi
		X	Ödeme kontrol listesi
		E	Proje uygun maliyetleri
		R	Proje uygun maliyetleri
		X	Proje uygun maliyetleri
		E	Ön ödeme risk puanı
		R	Ön ödeme risk puanı
		X	Ön ödeme risk puanı
		E	Sözleşme dosyası
		R	Sözleşme dosyası
		X	Sözleşme dosyası
		E	Iş kuralları
		R	Iş kuralları
		X	Iş kuralları
		E	Ön ödeme miktarı
		W	Ön ödeme miktarı
		E	Ödeme planı
		W	Ödeme planı
		E	Onay isteği
		W	Onay isteği
2	Ön ödeme miktarı listelenmesi	E	Ön ödeme miktarı
		R	Ön ödeme miktarı
		X	Ön ödeme miktarı
	Ön ödeme miktarı güncellenmesi	E	Ön ödeme miktarı
		W	Ön ödeme miktarı

FP#	FP name	Data Mov.	OOI/Data Group
	Ön ödeme miktarı silinmesi	E	Ön ödeme miktarı
		W	Ön ödeme miktarı
3	Ödeme oranlarının onaylanması	E	Onay isteği görüntüleme
		E	Onay istekleri
		R	Onay istekleri
		X	Onay istekleri
		E	Onay isteği
		R	Onay isteği
		X	Onay isteği
		E	Ön ödeme miktarı
		R	Ön ödeme miktarı
		X	Ön ödeme miktarı
		E	Ödeme planı
		R	Ödeme planı
		X	Ödeme planı
		E	Onay durumu
		W	Onay durumu
4	Onay durumu listelenmesi	E	Onay durumu
		R	Onay durumu
		X	Onay durumu
	Onay durumu güncellenmesi	E	Onay durumu
		W	Onay durumu
	Onay durumu silinmesi	E	Onay durumu
		W	Onay durumu
5	Başvuru sahibinden ön ödeme talebi alınması	E	Talep alınması isteği
		E	Ödeme talebi şartları
		R	Ödeme talebi şartları
		X	Ödeme talebi şartları
		E	Ön ödeme talebi

FP#	FP name	Data Mov.	OOI/Data Group
		W	Ön ödeme talebi
		E	Iş kuralları
		R	Iş kuralları
		X	Iş kuralları
		E	Teminat
		W	Teminat
6	Teminatın listelenmesi	E	Teminat
		R	Teminat
		X	Teminat
	Teminatın güncellenmesi	E	Teminat
		W	Teminat
	Teminatın silinmesi	E	Teminat
		W	Teminat
7	Ödeme talebinin listelenmesi	E	Ödeme talebi
		R	Ödeme talebi
		X	Ödeme talebi
	Ödeme talebinin güncellenmesi	E	Ödeme talebi
		W	Ödeme talebi
	Ödeme talebinin silinmesi	E	Ödeme talebi
		W	Ödeme talebi
8	Ön ödeme miktarının hesap numarasına aktarılması	E	Aktarım isteği
		E	Proje
		R	Proje
		X	Proje
		E	Ön ödeme miktarı
		R	Ön ödeme miktarı
		X	Ön ödeme miktarı
		E	Ön ödeme miktarı talimatı
		W	Ön ödeme miktarı talimatı

FP#	FP name	Data Mov.	OOI/Data Group
9	Hakediş esasına göre ön ödeme harcamalarını değerlendirme	E	Değerlendirme isteği
		E	Proje
		R	Proje
		X	Proje
		E	Eş finansman oranı
		E	Ön ödeme miktarı
		E	Ödeme talebi
		E	Ödeme planı
		E	Ara dönem hakediş miktarı
		E	Ödeme kontrol listesi
		R	Eş finansman oranı
		R	Ön ödeme miktarı
		R	Ödeme talebi
		R	Ödeme planı
		R	Ara dönem hakediş miktarı
		R	Ödeme kontrol listesi
		X	Eş finansman oranı
		X	Ön ödeme miktarı
		X	Ödeme talebi
		X	Ödeme planı
		X	Ara dönem hakediş miktarı
		X	Ödeme kontrol listesi
10	Ön ödemenin kapatılması	E	Kapatılması isteği
		E	Ödeme planı
		E	Ön ödeme miktarı
		E	Ara dönem hakediş miktarı
		E	Eş finansman oranı
		R	Ödeme planı
		R	Ön ödeme miktarı
		R	Ara dönem hakediş miktarı
		R	Eş finansman oranı
		X	Ödeme planı
		X	Ön ödeme miktarı
		X	Ara dönem hakediş miktarı
		X	Eş finansman oranı

FP#	FP name	Data Mov.	OOI/Data Group
		W	Ödeme planı
11	Ön ödemede kalan miktarın belirlenmesi	E	Kalan miktar isteği
		E	Ön ödeme miktarı
		E	Ara dönem hakediş miktarı
		R	Ön ödeme miktarı
		R	Ara dönem hakediş miktarı
		X	Ön ödeme miktarı
		X	Ara dönem hakediş miktarı
		E	Hakediş gerçekleşmemiş miktar
		W	Hakediş gerçekleşmemiş miktar
12	Hakediş gerçekleşmemiş miktarın listelenmesi	E	Hakediş gerçekleşmemiş miktar
		R	Hakediş gerçekleşmemiş miktar
		X	Hakediş gerçekleşmemiş miktar
	Hakediş gerçekleşmemiş miktarın güncellenmesi	E	Hakediş gerçekleşmemiş miktar
		W	Hakediş gerçekleşmemiş miktar
	Hakediş gerçekleşmemiş miktarın silinmesi	E	Hakediş gerçekleşmemiş miktar
		W	Hakediş gerçekleşmemiş miktar
13	Ara ödeme miktarının oluşturulması	E	Ödeme miktarı isteği
		E	Proje
		R	Proje
		X	Proje
		E	Eş finansman oranı
		E	Başvuru rehberi
		E	Proje uygun maliyetleri
		E	Ödeme planı

FP#	FP name	Data Mov.	OOI/Data Group
		R	Eş finansman oranı
		R	Başvuru rehberi
		R	Proje uygun maliyetleri
		R	Ödeme planı
		X	Eş finansman oranı
		X	Başvuru rehberi
		X	Proje uygun maliyetleri
		X	Ödeme planı
		W	Ödeme planı
		E	Onay isteği
		W	Onay isteği
14	Onay isteği listelenmesi	E	Onay isteği
		R	Onay isteği
		X	Onay isteği
	Onay isteği güncellenmesi	E	Onay isteği
		W	Onay isteği
	Onay isteği silinmesi	E	Onay isteği
		W	Onay isteği
15	Ara dönem ödeme miktarının onaylanması	E	Onama isteği
		E	Onay istekleri listesi
		R	Onay istekleri listesi
		X	Onay istekleri listesi
		E	Onay
		R	Onay
		X	Onay
		E	Ara dönem ödeme miktarı
		E	Ödeme planı
		R	Ara dönem ödeme miktarı
		R	Ödeme planı
		X	Ara dönem ödeme miktarı
		X	Ödeme planı
16	Onay durumu listelenmesi	E	Onay durumu
		R	Onay durumu

FP#	FP name	Data Mov.	OOI/Data Group
		X	Onay durumu
	Onay durumu güncellenmesi	E	Onay durumu
		W	Onay durumu
	Onay durumu silinmesi	E	Onay durumu
		W	Onay durumu
17	Ara ödeme miktarının hesap numarasına aktarılması	E	Aktarım talebi
		E	Proje
		R	Proje
		X	Proje
		E	Ara ödeme miktarı
		R	Ara ödeme miktarı
		X	Ara ödeme miktarı
		E	Ara ödeme miktarı talimatı
		W	Ara ödeme miktarı talimatı
18	Proje toplam uygun maliyetlerin belirlenmesi	E	Nihai maliyet belirleme isteği
		E	Proje
		R	Proje
		X	Proje
		E	Ön ödeme miktarı
		E	Ara dönem ödeme miktarı
		E	Proje uygun maliyetleri
		E	Temel proje raporları
		E	Sözleşme dosyası
		E	Ödeme kontrol listesi
		R	Ön ödeme miktarı
		R	Ara dönem ödeme miktarı
		R	Proje uygun maliyetleri
		R	Temel proje raporları
		R	Sözleşme dosyası
		R	Ödeme kontrol listesi
		X	Ön ödeme miktarı
		X	Ara dönem ödeme miktarı

FP#	FP name	Data Mov.	OOI/Data Group
		X	Proje uygun maliyetleri
		X	Temel proje raporları
		X	Sözleşme dosyası
		X	Ödeme kontrol listesi
		E	Proje nihai destek miktarı
		W	Proje nihai destek miktarı
19	Proje nihai destek miktarı listelenmesi	E	Proje nihai destek miktarı
		R	Proje nihai destek miktarı
		X	Proje nihai destek miktarı
	Proje nihai destek miktarı güncellenmesi	E	Proje nihai destek miktarı
		W	Proje nihai destek miktarı
	Proje nihai destek miktarı silinmesi	E	Proje nihai destek miktarı
		W	Proje nihai destek miktarı
20	Eş finansman oranında nihai destek miktarının indirilmesi	E	Indirim isteği
		E	Başvuru rehberi
		R	Başvuru rehberi
		X	Başvuru rehberi
		E	Proje nihai destek miktarı
		R	Proje nihai destek miktarı
		X	Proje nihai destek miktarı
		W	Proje nihai destek miktarı
21	Nihai ödemenin belirlenmesi	E	Nihai ödeme isteği
		E	Proje
		R	Proje
		X	Proje
		E	Business rule
		R	Business rule
		X	Business rule
		E	Proje nihai destek miktarı

FP#	FP name	Data Mov.	OOI/Data Group
		E	Ön ödeme miktarı
		E	Ara dönem ödeme miktarı
		E	Ödeme planı
		R	Proje nihai destek miktarı
		R	Ön ödeme miktarı
		R	Ara dönem ödeme miktarı
		R	Ödeme planı
		X	Proje nihai destek miktarı
		X	Ön ödeme miktarı
		X	Ara dönem ödeme miktarı
		X	Ödeme planı
		W	Ödeme planı
		E	Onay isteği
		W	Onay isteği
		E	Proje nihai ödeme miktarı
		W	Proje nihai ödeme miktarı
22	Proje nihai ödeme miktarı listelenmesi	E	Proje nihai destek miktarı
		R	Proje nihai destek miktarı
		X	Proje nihai destek miktarı
	Proje nihai ödeme miktarı güncellenmesi	E	Proje nihai destek miktarı
		W	Proje nihai destek miktarı
	Proje nihai ödeme miktarı silinmesi	E	Proje nihai destek miktarı
		W	Proje nihai destek miktarı
23	Nihai ödeme miktarının onaylanması	E	Onay
		E	Onay istekleri
		R	Onay istekleri
		X	Onay istekleri
		E	Onay isteği
		E	Ödeme planı
		E	Proje nihai ödeme miktarı
		R	Onay isteği
		R	Ödeme planı

FP#	FP name	Data Mov.	OOI/Data Group
		R	Proje nihai ödeme miktarı
		X	Onay isteği
		X	Ödeme planı
		X	Proje nihai ödeme miktarı
		E	Onay durumu
		W	Onay durumu
24	Onay durumu listelenmesi	E	Onay durumu
		R	Onay durumu
		X	Onay durumu
	Onay durumu güncellenmesi	E	Onay durumu
		W	Onay durumu
	Onay durumu silinmesi	E	Onay durumu
		W	Onay durumu
25	Nihai ödeme miktarının hesap numarasına aktarılması	E	Hesaba aktarım isteği
		E	Proje
		R	Proje
		X	Proje
		E	Business rule
		R	Business rule
		X	Business rule
		E	Proje nihai ödeme miktarı
		R	Proje nihai ödeme miktarı
		X	Proje nihai ödeme miktarı
		E	Ödeme talimatı
		W	Ödeme talimatı
26	Kalan eksi değerin ajans hesabına aktarılması	E	Hesaba aktarım isteği
		E	Proje
		R	Proje
		X	Proje
		E	Proje nihai ödeme miktarı
		R	Proje nihai ödeme miktarı

FP#	FP name	Data Mov.	OOI/Data Group
		X	Proje nihai ödeme miktarı
		E	Ajans hesabına geri aktarım talimatı
		W	Ajans hesabına geri aktarım talimatı
27	Yararlanıcıdan geri ödeme yapılması	E	Geri ödeme yapılması
		E	Proje geri ödeme miktarı
		R	Proje geri ödeme miktarı
		X	Proje geri ödeme miktarı
		E	Business rule
		R	Business rule
		X	Business rule
	Toplam CFP	318	

SORUN YÖNETİMİ SÜRECİ

Use Case	#1: Proje uygulama sorunu oluşturulması
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı
Minimal Guarantees	-
Success Guarantees	Sorun kaydedildi
Preconditions	-
Triggers	Sorun tespit edildi, yararlanıcıdan sorun geri bildirimi yapıldı
Main Scenario Success	1- İDB uzmanı sisteme girer 2- İDB uzmanı proje listesi görüntüleme isteğinde bulunur 3- Sistem projeleri görüntüler 4- İDB uzmanı proje uygulama sorunu oluşturur 5- Sistem proje uygulama sorununu kayıt eder. 6- İDB uzmanı proje uygulama sorunu ve sözleşme dosyasını görüntüleme isteğinde bulunur. 7- Sistem proje uygulama sorunu ve sözleşme dosyasını görüntüler 8- İDB uzmanı proje uygulama sorunu raporunu oluşturur.

	9- Sistem proje uygulama sorunu raporunu kayıt eder.
Extensions	4a- İDB uzmanı proje uygulama sorunu listeleme isteğinde bulunur 1. Sistem proje uygulama sorunlarını listeler 4b- İDB uzmanı proje uygulama sorunu güncelleme isteğinde bulunur 1. Sistem değişiklikleri kayıt eder 4c- İDB uzmanı proje uygulama sorunu silme isteğinde bulunur 1. Sistem değişiklikleri kayıt eder 8a- İDB uzmanı proje uygulama sorunu raporu listeleme isteğinde bulunur 2. Sistem proje uygulama sorunu raporlarını listeler 8b- İDB uzmanı proje uygulama sorunu raporunu güncelleme isteğinde bulunur 2. Sistem değişiklikleri kayıt eder 8c- İDB uzmanı proje uygulama sorunu raporunu silme isteğinde bulunur 2. Sistem değişiklikleri kayıt eder

Use Case	#2: Sorun uyarı mekanizmasının oluşturulması
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı
Minimal Guarantees	-
Success Guarantees	Sorun uyarı mekaniması oluşturuldu
Preconditions	Sorun değerlendirildi
Triggers	Sorun değerlendirildi
Main Scenario Success	1- İDB uzmanı sisteme girer 2- İDB uzmanı Sorun uyarı mekanizması oluşturma isteğinde bulunur. 3- Sistem uyarı mekanizmaları seçeneğini listeler 4- İDB uzmanı sözlü sorun mekanizması oluşturma isteğinde bulunur 5- Sistem sözlü sorun mekanizması oluşturur.

Extensions	4a İDB uzmanı erken uyarı mekanizması oluşturma isteğinde bulnur 1. Sistem erken uyarı mekanizmasını kayıt eder 4b İDB uzmanı usulsüzlük mekanizması oluşturma isteğinde bulnur 1. Sistem usulsüzlük mekanizmasını kayıt eder.

Use Case	#3: Sözlü uyarı sonucunun takip edilmesi
Primary Actor	İDB uzmanı
Stakeholders	İDB uzmanı
Minimal Guarantees	-
Success Guarantees	Sorun belirlenen zamanda giderildi
Preconditions	Sözlü uyarı mekanizması oluşturuldu
Triggers	
Main Scenario Success	1- İDB uzmanı sisteme girer. 2- İDB uzmanı sözlü sorun takibi görüntüleme isteğinde bulunur 3- Sistem sözlü sorun takibini görüntüler 4- İDB uzmanı proje uygulama sorunu raporunu görüntüleme isteğinde bulunur 5- Sistem proje uygulama sorun raporunu görüntüler. 6- İDB uzmanı sorun aşamasını proje uygulama sorun raporuna girer. 7- Sistem proje uygulama sorunu raporunu kayıt eder.
Extensions	

Use Case	#4:Erken uyarıda izlenmesi gereken düzeltici tedbirlerin hazırlanması
Primary Actor	İzleme uzmanı
Stakeholders	İzleme uzmanı

Minimal Guarantees	-
Success Guarantees	Erken uyarı çalışmaları gerçekleştirildi.
Preconditions	Erken uyarı mekanizmasına karar verildi.
Triggers	
Main Scenario Success	1- İzleme uzmanı sisteme girer 2- İzleme uzmanı sorunlu projeleri görüntüleme isteğinde bulunur 3- Sistem sorunlu projeleri görüntüler 4- İzleme uzmanı proje uygulama sorunu ve proje uygulama sorunu raporu görüntüleme isteğinde bulunur. 5- Sistem proje uygulama sorunu ve proje uygulama sorunu raporunu görüntüler. 6- İzleme uzmanı proje uygulama sorun aşamasını değiştirir. 7- Sistem proje uygulama sorununu kayıt eder.
Extensions	-

Use Case	#5: Erken uyarı raporunun hazırlanması
Primary Actor	İzleme uzmanı
Stakeholders	İzleme uzmanı
Minimal Guarantees	-
Success Guarantees	Erken uyarı ön çalışmaları gerçekleştirildi.
Preconditions	Erken uyarı raporu hazırlandı
Triggers	
Main Scenario Success	1- İzleme uzmanı sisteme girer 2- İzleme uzmanı sorunlu projeleri görüntüleme isteğinde bulunur 3- Sistem sorunlu projeleri görüntüler 4- İzleme uzmanı erken uyarı raporu hazırlama isteğinde bulunur 5- Sistem proje uygulama sorun, proje uygulama sorunu raporu ve Ek I-13:erken uyarı raporunu görüntüler 6- İzleme uzmanı erken uyarı raporu oluşturur. 7- Sistem erken uyarı raporunu kayıt eder. 8- İzleme uzmanı erken uyarı raporu için onay isteği oluşturur. 9- Sistem erken uyarı raporunu kayıt eder.

Extensions	4a- izleme uzmanı erken uyarı raporu listeleme, güncelleme ve silme isteğinde bulunur. 8a- izleme uzmanı onay isteği listeleme, güncelleme ve silme isteğinde bulunur.

Use Case	#6: Erken uyarı raporunun onaylanması
Primary Actor	GS
Stakeholders	GS
Minimal Guarantees	-
Success Guarantees	Erken uyarı raporu onaylandı
Preconditions	
Triggers	
Main Scenario Success	1- GS sisteme girer 2- GS onay isteklerini görüntüleme isteğinde bulunur 3- Sistem onay isteklerini görüntüler 4- GS onay isteği deyatlarını görüntüleme isteğinde bulnur 5- Sistem onay isteği detaylarını görüntüler 6- GS erken uyarı raporunu görüntüleme isteğinde bulunur 7- Sistem erken uyarı raporunu görüntüler 8- GS anlık izleme ziyareti oluşmaksızın onay durumu oluşturur. 9- Sistem onay durumunu kayıt eder
Extensions	8a- GS anlık izleme durumu oluşturur 1. Sistem anlık izleme durumunu kayıt eder 2. İzleme uzmanı include: ANLIK İZLEME sonucunu kayıt eder. 8b- GS onay durumunu listeleme, güncellemek ve silme isteğinde bulunur.

Use Case	#7: Erken uyarı raporu sorun giderilmesi
Primary Actor	İzleme uzmanı
Stakeholders	İzleme uzmanı
Minimal Guarantees	-
Success Guarantees	Sorun giderildi
Preconditions	Erken uyarı raporu oluşturuldu
Triggers	
Main Scenario Success	1- İzleme uzmanı sisteme girer 2- İzleme uzmanı Erken uyarı raporu görüntüleme isteğinde bulunur 3- Sistem erken uyarı raporunu görüntüler 4- İzleme uzmanı erken uyarı raporunu yararlanıcıya gönderir. 5- sistem yararlanıcıya gönderildi bilgisini kayıt eder. 6- İzleme uzmanı erken uyarı durumunu takip eder. 7- Sistem yararlanıcı erken uyarı durum bilgisini görüntüler. 8- İzleme uzmanı sorun giderildi raporunu oluşturur.
Extensions	7b – sistem erken uyarı sorunu düzeltilememe sebebini görüntüler 1. İzleme uzmanı erken uyarı durumunu görüntüleme isteğinde bulunur 2. Sistem erken uyarı durumunu görüntüler 3. İzleme uzmanı erken uyarı raporu sorun bilgisini oluşturur 4. Sistem erken uyarı sorun bilgisini kayıt eder 5. İzleme uzmanı erken uyarı onay isteği oluşturur 5a- izleme uzmanı erken uyarı onayı listeleme, güncelleme ve silme isteğinde bulunur. 1. Sistem değişiklikleri kayıt eder. 6. Sistem onay isteğini kayıt eder.

Use Case	#8: Sorun ile ilgili karar verilmesi
Primary Actor	GS
Stakeholders	GS
Minimal Guarantees	-
Success Guarantees	Erken uyarı işlemleri tamamlandı
Preconditions	Ara veya nihai rapor incelenmesi tamamlandı
Triggers	Proje ara ödemesi gerçekleşti
Main Scenario Success	1- GS sisteme girer 2- GS erken uyarı raporu onay isteklerini görüntüler 3- Sistem erken uyarı raporu onay isteklerini görüntüler 4- GS düzeltici tedbirlere devam edilmesi durumunu oluşturur 5- Sistem düzeltici tedbir durumunu kayıt eder
Extensions	4a- GS usulsüzlük işlemleri mekanizması durumunu oluşturur 1- izleme uzmanı include : USULSÜZLÜK MEKANİZMASI 2- Sistem kayıt eder 4b- GS sözleşmenin Feshi durumunu oluşturur. 1- İzleme uzmanı include : SÖZLEŞME FESHİ 4c- GS ödemenin durdurulması durumunu oluşturur. 1- GS sözleşme durumunu görüntüleme isteğinde bulunur 2- Sistem sözleşme durumunu görüntüler. 3- GS sözleşme durumuna ödemenin durdurulmasını girer 4- Sistem ödemenin durdurulması durumunu kayıt eder.

Use Case	#9: Usulsüzlük raporunun hazırlanması
Primary Actor	izleme uzmanı
Stakeholders	izleme uzmanı

Minimal Guarantees	-
Success Guarantees	Usulsüzlük raporu hazırlandı
Preconditions	Usulsüzlük mekanizması uygun görüldü.
Triggers	
Main Scenario Success	1- izleme uzmanı sisteme girer 2- izleme uzmanı usulsüzlük işlemlerine dair projeleri görüntüler 3- sistem usulsüzlük işlemi projesini görüntüler 4- izleme uzmanı proje uygulama sorun raporunu, proje uygulama sorun ve Ek I-15: usulsüzlük raporunu görüntüleme isteğinde bulunur. 5- Sistem proje uygulama sorun raporunu, proje uygulama sorun ve Ek I-15: usulsüzlük raporunu görüntüler 6- izleme uzmanı usulsüzlük raporu oluşturur 7- sistem usulsüzlük raporunu kayıt eder. 8- izleme uzmanı anlık izleme oluşturmaz 9- sistem anlık izleme oluşturulmadığını kayıt eder
Extensions	8a izleme uzmanı anlık izleme oluşturur include: ANLIK İZLEME use case 1. sistem sonucu kayıt eder

Use Case	#2: Usulsüzlük durumunun değerlendirilmesi
Primary Actor	GS
Stakeholders	GS
Minimal Guarantees	-
Success Guarantees	-
Preconditions	-
Triggers	-
Main Scenario Success	1- GS sisteme girer 2- GS usulsuzluk belirlenen projeleri görütüleme isteğinde bulunur 3- Sistem usulsüz projeleri görüntüler 4- GS düzeltici tedbirlerin alınması isteğini yaratır 5- Sistem düzeltici tedbir isteğini kayıt eder
Extensions	4a- GS usulsüzlük durumunu üptal eder 1- Sistem usulsüzlük durumunu iptal edip, projenin usulsüzlük değerini sıfırlar.

	4b- GS sözleşmenin feshine karar verir 1- Sistem sözleşme fesih durumunu kayıt eder

Use Case	#9: İzlenmesi gereken düzeltici tedbirlerin alınması
Primary Actor	izleme uzmanı
Stakeholders	izleme uzmanı
Minimal Guarantees	-
Success Guarantees	İzlenmesi gereken düzeltici tedbirler hazırlandı.
Preconditions	Durumun düzeltici tedbirlerle düzeltilmesine karar verildi
Triggers	
Main Scenario Success	1- İzleme uzmanı sisteme girer 2- İzleme uzmanı usulsüz projeleri görüntüleme isteğinde bulunur 3- Sistem usulsüz projeleri görüntüler 4- İzleme uzmanı izlenmesi gereken düzeltici tedbirleri hazırlama isteğinde bulunur 5- Sistem proje uygulama sorun ve proje uygulama sorunu raporunu görüntüler. 6- İzleme uzmanı proje uygulama sorun'a düzeltici tedbirleri ekler 7- Sistem "proje uygulama sorun" u kayıt eder.
Extensions	6a- izleme uzmanı proje uygulama sorun'u görüntüleme isteğinde bulunur. 1. Sistem düzeltici tedbir eklenmiş proje uygulama sorun'unu görüntüler 2. İzleme uzmanı sorun giderildi bilgisini ekler 2a- sorun giderilmedi bilgisi eklenir 1. GS uzmanı sözleşmeyi fesh eder. 3. Sistem usulsüzlük işlemlerini tamamlar.

FP#	FP name	Data Mov.	OOI/Data Group
1	Proje uygulama sorunu oluşturulması	E	Sorun oluşturma isteği
		E	Proje listesi
		R	Proje listesi
		X	Proje listesi
		E	Proje

FP#	FP name	Data Mov.	OOI/Data Group
		R	Proje
		X	Proje
		E	Proje uygulama sorun
		W	Proje uygulama sorun
2	Proje uygulama sorunu raporu oluşturulması	E	Sorun raporu isteği
		E	Proje
		R	Proje
		X	Proje
		E	Business rule
		R	Business rule
		X	Business rule
		E	Proje uygulama sorun
		E	Sözleşme dosyası
		R	Proje uygulama sorun
		R	Sözleşme dosyası
		X	Proje uygulama sorun
		X	Sözleşme dosyası
		E	Proje uygulama sorunu raporu
		W	Proje uygulama sorunu raporu
3	Proje uygulama sorun raporu için listeleme, güncelleme ve silme	7 DM	
4	Proje uygulama sorun için listeleme, güncelleme ve silme	7 DM	
5	Sorun uyarı mekanizmasının oluşturulması	E	Sorun uyarı isteği
		E	Sorun uyarı mekanizmları
		R	Sorun uyarı mekanizmları
		X	Sorun uyarı mekanizmları
		E	Sorun tip belirleme
		W	Sorun tip belirleme
6	Sözlü uyarı sonucunun	E	Sözlü uyarı sonuç takip isteği

FP#	FP name	Data Mov.	OOI/Data Group
	izlenmesi		
		E	Sözlü sorun listesi
		R	Sözlü sorun listesi
		X	Sözlü sorun listesi
		E	Proje uygulama sorun
		E	Proje uygulama sorunu rapor
		R	Proje uygulama sorun
		R	Proje uygulama sorunu rapor
		X	Proje uygulama sorun
		X	Proje uygulama sorunu rapor
		W	Proje uygulama sorunu rapor
7	Erken uyarıda izlenmesi gereken düzeltici tedbirlerin hazırlanması	E	Düzeltici tedbirler isteği
		E	Sorunlu proje
		R	Sorunlu proje
		X	Sorunlu proje
		E	Proje uygulama sorun
		E	Proje uygulama sorun rapor
		R	Proje uygulama sorun
		R	Proje uygulama sorun rapor
		X	Proje uygulama sorun
		X	Proje uygulama sorun rapor
		W	Proje uygulama sorun
8	Erken uyarı raporunun hazırlanması	E	Erken uyarı rapor isteği
		E	Sorunlu proje
		R	Sorunlu proje
		X	Sorunlu proje
		E	Iş kuralları
		R	Iş kuralları
		X	Iş kuralları
		E	Proje uygulama sorun
		E	Proje uygulama sorunu rapor
		E	Ek-13 erken uyarı raporu
		R	Proje uygulama sorun
		R	Proje uygulama sorunu rapor

FP#	FP name	Data Mov.	OOI/Data Group
		R	Ek-13 erken uyarı raporu
		X	Proje uygulama sorun
		X	Proje uygulama sorunu rapor
		X	Ek-13 erken uyarı raporu
		E	Erken uyarı raporu
		W	Erken uyarı raporu
9	Erken uyarı raporunun onaya sunulması	E	Onay isteği
		E	Erken uyarı raporu
		R	Erken uyarı raporu
		X	Erken uyarı raporu
		E	Onay isteği
		W	Onay isteği
10	Erken uyarı raporu listeleme, güncelleme ve silme	7 DM	
11	Onay isteği listeleme, güncelleme ve silme	7 DM	
12	Erken uyarı raporunun onaylanması	E	Onaylama
		E	Onay istekleri
		R	Onay istekleri
		X	Onay istekleri
		E	Onay detayları
		R	Onay detayları
		X	Onay detayları
		E	Iş kuralları
		R	Iş kuralları
		X	Iş kuralları
		E	Erken uyarı raporu
		E	Onay isteği
		R	Erken uyarı raporu
		R	Onay isteği
		X	Erken uyarı raporu
		X	Onay isteği
		E	Onay durumu

FP#	FP name	Data Mov.	OOI/Data Group
		W	Onay durumu
13	Onay durumu listeleme, güncelleme ve silme	7 DM	
14	Anlık izleme ziyareti oluşturma	E	Anlık izleme
		W	Anlık izleme
		W	Proje
15	Erken uyarı raporu sorun giderilmesi	E	Erken uyarı gönderim isteği
		E	Erken uyarı raporu
		R	Erken uyarı raporu
		X	Erken uyarı raporu
		E	Erken uyarı-proje
		W	Erken uyarı-proje
16	Erken uyarı düzeltilememe durumu	E	Erken uyarı düzeltme isteği
		E	Erken uyarı durumu
		R	Erken uyarı durumu
		X	Erken uyarı durumu
		E	Erken uyarı raporu
		R	Erken uyarı raporu
		X	Erken uyarı raporu
		W	Erken uyarı raporu
		E	Onay isteği
		W	Onay isteği
17	Onay isteği listeleme, güncelleme ve silme	7 DM	
18	Düzeltici tedbirlere devam edilmesi	E	Karar isteği
		E	Erken uyarı onay isteği
		R	Erken uyarı onay isteği
		X	Erken uyarı onay isteği
		E	Proje uygulama sorun-düzeltici tedbirlere devam

FP#	FP name	Data Mov.	OOI/Data Group
		W	Proje uygulama sorun-düzeltici tedbirlere devam
19	Usulsüzlük mekanizması oluşturma	E	Usulsüzlük isteği
		E	Proje uygulama sorun-usulsüzlük
		W	Proje uygulama sorun-usulsüzlük
20	Sözleşme feshi	E	Sözleşme feshi isteği
		E	Proje uygulama sorun-sözleşme feshi
		W	Proje uygulama sorun-sözleşme feshi
21	Ödemenin durdurulması	E	Sözleşme-ödeme durdurma isteği
		E	Sözleşme durumu
		R	Sözleşme durumu
		X	Sözleşme durumu
		W	Sözleşme durumu
22	Usulsüzlük raporunun hazırlanması	E	Usulsüzlük rapor isteği
		E	Usulsüz-proje
		R	Usulsüz-proje
		X	Usulsüz-proje
		E	Proje uygulama sorunu raporu
		E	Proje uygulama sorun
		E	Ek-15 usulsüzlük raporu
		R	Proje uygulama sorunu raporu
		R	Proje uygulama sorun
		R	Ek-15 usulsüzlük raporu
		X	Proje uygulama sorunu raporu
		X	Proje uygulama sorun
		X	Ek-15 usulsüzlük raporu
		E	Iş kuralları
		R	Iş kuralları

FP#	FP name	Data Mov.	OOI/Data Group
		X	Iş kuralları
		E	Usulsüzlük raporu
		W	Usulsüzlük raporu
23	Usulsüzlük anlık izleme oluşturma	E	Anlık izleme
		E	Usulsüzlük rapor-anlık izleme
		W	Usulsüzlük rapor-anlık izleme
24	Usulsüzlük raporu listeleme, güncelleme ve silme	7 DM	
25	Usulsüzlük durum değerlendirme	E	Usulsüzlük durum değerlendirme
		E	Usulsüzlük durumu-proje
		R	Usulsüzlük durumu-proje
		X	Usulsüzlük durumu-proje
		E	Düzeltici tedbir
		E	Usulsüzlük iptal
		E	Sözleşme iptal
		W	Düzeltici tedbir
		W	Usulsüzlük iptal
		W	Sözleşme iptal
26	Durum düzeltici tedbirler alınması	E	Durum düzeltme
		E	Usulsüz-proje
		R	Usulsüz-proje
		X	Usulsüz-proje
		E	Proje uygulama sorun
		E	Proje uygulama sorun raporu
		R	Proje uygulama sorun
		R	Proje uygulama sorun raporu
		X	Proje uygulama sorun
		X	Proje uygulama sorun raporu
		W	Proje uygulama sorun
27	Düzeltici tedbir eklenmiş sözleşme feshi	E	Fesih

FP#	FP name	Data Mov.	OOI/Data Group
		E	Proje uygulama sorun
		R	Proje uygulama sorun
		X	Proje uygulama sorun
		W	Proje uygulama sorun
	Toplam CFP	226	