Vol.5 No.2

Advanced Engineering and ICT-Convergence Proceedings (AEICP)

9th International Conference on Advanced Engineering and ICT-Convergence (ICAEIC-2022)

July 13-15, 2022

Organized by

ICT - Advanced Engineering Society,
Seoul, Korea (ICT-AES)
Bima Build. #525, 20 Kwangwoon-ro, Nowon-gu, Seoul, Korea
Email: info@ictaes.org, Tel.: +82-2-940-8626 / 8637

Sponsored by

TRACOM

BonC nnovators

Cost Extraction with Reverse Engineering Approach

So Young Moon[1], and R. Young Chul Kim*

[1,]*Dept. of Software and Communication Engineering, SE Lab., Hongik University

{whit2[1], bob*}@hongik.ac.kr

Abstract

In the initial stage of a software development project, it is difficult to estimate the size of the software without any information about the new software project. But still, we use the function point approach to estimate cost of a new project. Also, we can't prove it whether it is correct the cost before and after the project. So far, in real software fields, everybody has only focused on estimating the cost of calculating the size of a project using the Function Point, but still not proved it. To solve this problem, we propose our cost extraction approach based on a reverse engineering approach that proves it after the project. This approach tries to prove the cost with the number of implemented FP bewteen before and after a project.

Keywords: *Function Point, Reverse Engineering, Extracting Cost with Reverse Engineeing, Code Visualization.*

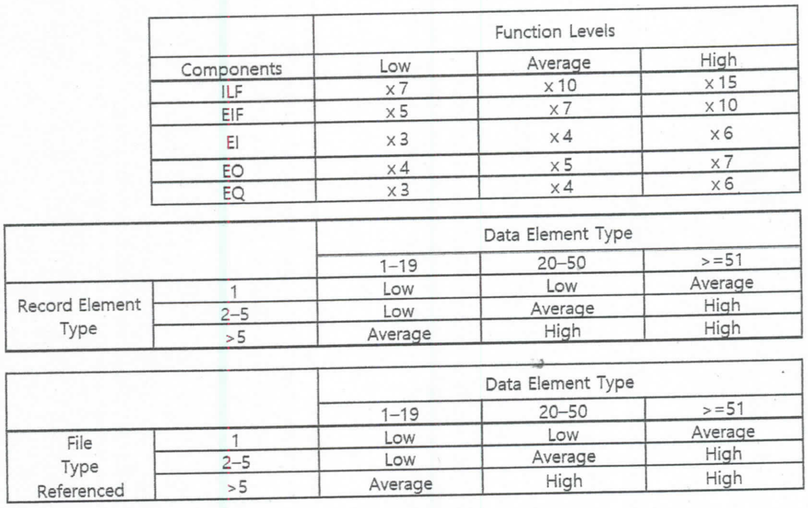
1. Introduction

As software is used in many fields, the cost estimation of a new project is a very important issue. From the point of view of the order holder, the problems of software size calculation are as follows: 1) there are difficulties in judging/estimating the size due to the unclear and incomplete requirements, 2) some incorrect requirements make incorrect cost estimation [1, 2], and 3) it is difficult for the contracting company to evaluate the adequacy of the scope of tasks compared to the project cost, and to judge whether the client's task is changed or additional tasks are justified during project execution. From the point of view of the ordering party, they require any proof that the purchaser has not paid excessively, but impossible to do it. In some studies, the function score is improved to predict the development cost [3, 4]. Still proving to spend a reasonable cost on the source code has not been performed. To solve this problem, cost verification should be performed based on the most recently developed source code.

The structure of this paper is as follows. Chapter 2 describes related research, and Chapter 3 explains the verification method based on the reverse engineering method with an application example. Chapter 4 describes the conclusions.

Advanced Engineering and ICT-Convergence Proceedings (AEICP)
ISSN : 2635-4586

2. Related Work



	Function Levels		
Components	Low	Average	High
ILF	x 7	x 10	x 15
EIF	x 5	x 7	x 10
EI	x 3	x 4	x 6
EO	x 4	x 5	x 7
EQ	x 3	x 4	x 6

		Data Element Type		
		1–19	20–50	>=51
Record Element Type	1	Low	Low	Average
	2–5	Low	Average	High
	>5	Average	High	High

		Data Element Type		
		1–19	20–50	>=51
File Type Referenced	1	Low	Low	Average
	2–5	Low	Average	High
	>5	Average	High	High

Figure 1. Function Point's Complexity.

Figure 1 shows the function point's complexity by the IPFG as follows. The function score is the sum of the function complexity score calculated from the data function type and the transaction function type. As a data function type, a file used inside the application boundary is called an internal logical file (ILF), and when a file of another application is used, it is called an external interface file (EIF). We calculate the complexity for ILF and EIF in a specific system analysis. The function point is calculated by summing the result of calculating the data function point and the transaction function point. The transaction function score is calculated with the complexity for EI, EO, and EQ. The transaction function type is classified as external input when data processing is performed inside or outside the application boundary, external input when there is input processing, external inquiry when there is data inquiry, and external output when there is a file creation or calculation between data [5].

3. Cost Extraction Based on Reverse Engineering

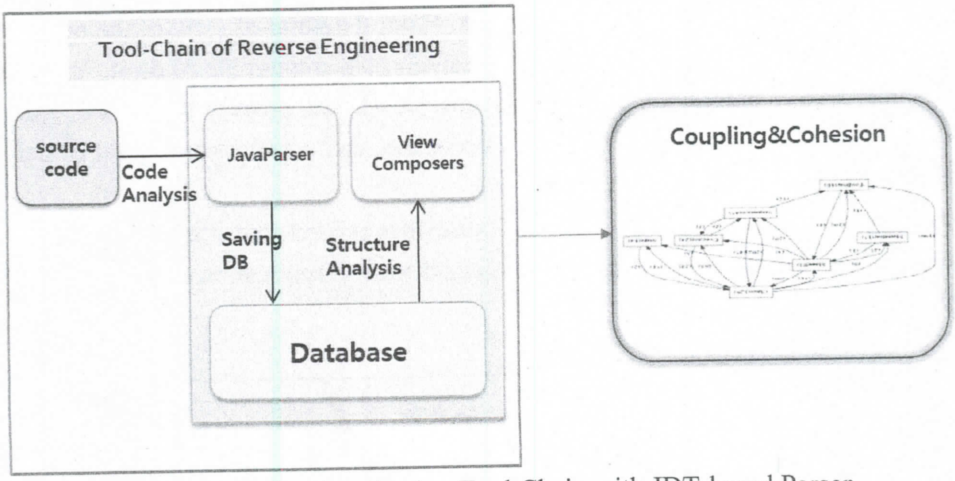
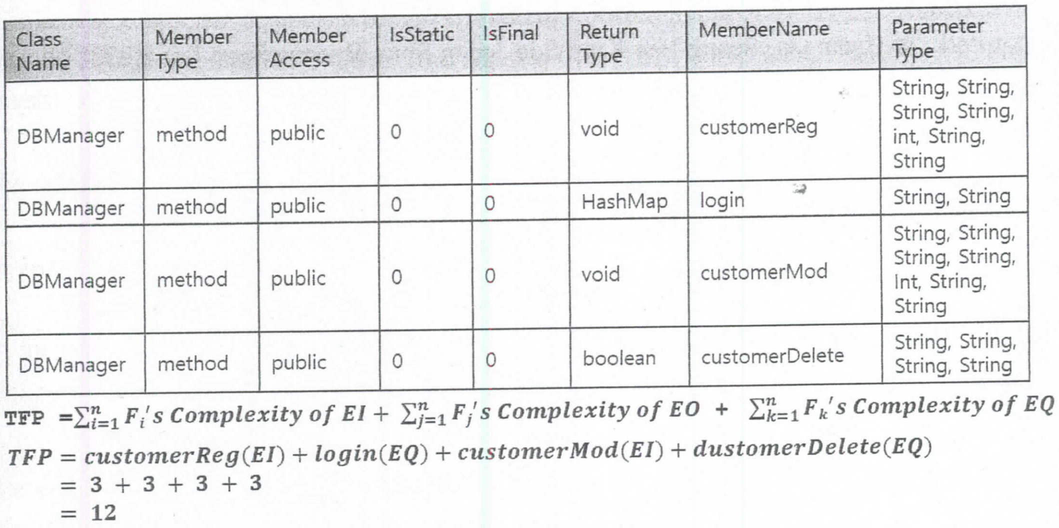


Figure 2. Code Visualization Tool Chain with JDT-based Parser.

Figure 2 shows the tool chain for code visualization based on reverse engineering. The execution process of the tool chain is as follows: 1) Java parser parses the source code, 2) The parsed code sentence is stored in the database, 3) Select data from the database by creating query statements for ILF, EIF, EI, EO, and EQ related to the function point, and create a Dot script using the searched content, and 4) Visualization is performed with the contents of the Dot script based on the query statement created in (3) [6].



Class Name	Member Type	Member Access	IsStatic	IsFinal	Return Type	MemberName	Parameter Type
DBManager	method	public	0	0	void	customerReg	String, String, String, String, int, String, String
DBManager	method	public	0	0	HashMap	login	String, String
DBManager	method	public	0	0	void	customerMod	String, String, String, String, Int, String, String
DBManager	method	public	0	0	boolean	customerDelete	String, String, String, String

$$\mathbf{TFP} = \sum_{i=1}^{n} F_i{}'s\ Complexity\ of\ EI + \sum_{j=1}^{n} F_j{}'s\ Complexity\ of\ EO + \sum_{k=1}^{n} F_k{}'s\ Complexity\ of\ EQ$$

$$\begin{aligned} TFP &= customerReg(EI) + login(EQ) + customerMod(EI) + dustomerDelete(EQ) \\ &= 3 + 3 + 3 + 3 \\ &= 12 \end{aligned}$$

Figure 3. Various Methods in Source Code.

Figure 3 is a part of DBManger class in DB. Various Methods such as customerReg, login, customerMod, and customerDelete exist in the DBManager class. Information such as method return type, parameter type, member access specifier, static, final, etc. can be found. Get the name and information of the method in the code from the contents of this table. The formula in Figure 3 is explained as follows. EI, EO, and EQ are classified by function, and the complexity is calculated according to the number of RET and FTR. Therefore, customerReg is EI and complexity is 3. login is EQ and complexity is 3. customerMod is EI and complexity is 3. customerDelete is EQ and complexity is 3. The function score is 12 in Figure 3.

4. Conclusion

Estimating the cost or size of a software development project is critical to the project's success. This study attempts to prove project size estimation by analyzing data and transaction functions through reverse engineering. The purpose of this study is to verify the size of the project at the time of ordering and completion.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2021R1I1A1A01044060), and also supported by the Ministry of Education and the Korea Research Foundation (F21YY8102068) and the government (Ministry of Education) with the support of the Korea Research Foundation (No. 2021R1I1A305040711) in 2022.

References

[1] Hyeon Seung Lee, Yun Sun Lee(2017), The Current Status, Problems, and Ways of Improving the Management of Public SW Project Contracting, No.2017-002, SPRi Insight Report(pp.7-8)

[2] NIPA(2021), Practice Guidelines for Detailing Requirements for writing Public SW Project Request for Proposal.

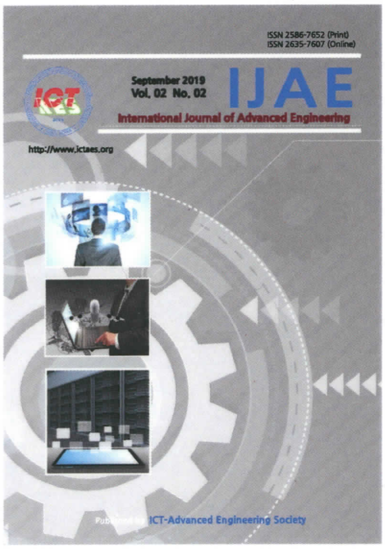
[3] Chan Gyu Park, Ja Hwan Gu, Seong Hee Kim, Soo Jeong Shin, Byeong Seon Song(2002.November).A Study on the Estimation of Software Development Cost of IT Projects in Public Sector. Management Science, Vol. 19, No. 2, pp.191-204

[4] Yeon Shik Ahn(2009). An Enhanced Function Point Model for Software Size Estimation: Micro-FP Model.Vol. 14, No. 12, (pp.225-232). Korea Society of Computer Information.

[5] Manfred Bundschuh, Carol Dekkers(2008), The IFPUG Function Point Counting Method, The IT Measurement Compendium, Springer.

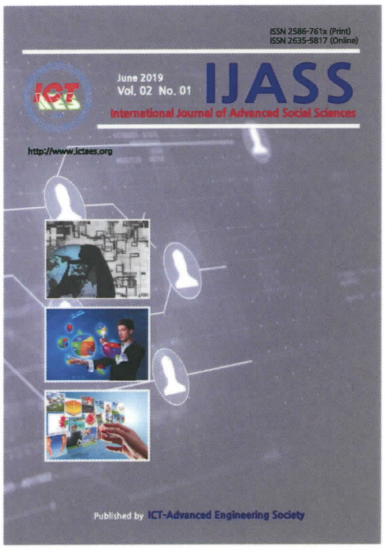
[6] So Young Moon, R. Young Chul Kim(2015). Code Structure Visualization with a Tool-Chain Method. International Journal of Applied Engineering Research (pp.214-218). Vol. 10. No.90. Research India Publications.

ICT-AES Publications



International Journal of Advanced Engineering (IJAE) is a scholarly open access, a peer-reviewed, and half-yearly journal focusing on theories, methods, and applications in Engineering and Technology. It covers all areas of Engineering and Technology, publishing original research articles and technical notes. All manuscript that are submitted must report unpublished work and cannot be under consideration for publication elsewhere.

Print ISSN: 2586-7652 | Online ISSN: 2635-7607


ISSN 2586-761x (Print)
ISSN 2635-5817 (Online)
June 2019
Vol. 02 No. 01
IJASS
International Journal of Advanced Social Sciences
http://www.ictaes.org
Published by ICT-Advanced Engineering Society

International Journal of Advanced Social Sciences (IJASS) is a scholarly open access, Peer-reviewed, and half-yearly journal focusing on theories, methods, and applications in Social Sciences. It publishes both theoretical and empirical articles and case studies relating to sociology, political science, history, a law in society and related disciplines. Published articles use scientific research methods, including statistical analysis, case studies, field research and historical analysis. All manuscript that are submitted must report unpublished work and cannot be under consideration for publication elsewhere.

Print ISSN:2586-761x | Online ISSN: 2635-5817

All submitted articles should report original, previously unpublished research results, experimental or theoretical and will be peer-reviewed.