Industrial Engineering Journal
ISSN: 0970-2555
Volume : 53, Issue 5, No.7, May : 2024

ANALYZING THE COMPLEXITY ADJUSTMENT FACTORS OF AN INFORMATION SYSTEM USING FUNCTION POINT

Ms.Akanksha Singh, Student, Dept.of CSE-AIML, ABES Engineering College, Ghaziabad, India.
Ms.Anu Singh, Student, Dept.of CSE-AIML, ABES Engineering College, Ghaziabad, India.
Dr.Tanveer Hassan, Assistant Professor, Dept.of CSE-AIML, ABES Engineering College, Ghaziabad, India.

ABSTRACT

Function point is applied to measure the functional size of an information system. During the computational process of function point, first the given information system is evaluated based on five parameters, i.e., input to the system, output of the system, queries of the system, internal logical files of the system, and external interface files of the system and after that it evaluated on the basis of seventeen complexity adjustment factors (CAFs). Based on our review, we found that these general system characteristics are not sufficient to compute the functional size of an information system because there are various non-functional requirements which are not the part of these CAFs like security, performance, usability etc. Therefore, to address this issue, new CAFs have been introduced in this article so that an information system can be analysed systematically based on security,cost and usability. Finally, the requirements of an institute examination system are analyzed using the extended CAFs

Keywords:
Function point, complexity adjustment factor, information system, institute examination system.

.

I. Introduction

The function point approach (FPA) was introduced by Allan Albrecht in 1979 for measuring application development productivity based on the outcome of project that was developed from 1974 to 1978 at International Business Machine (IBM) [1]. The objective of FPA is to measure the size of an information system from its functional requirements (FRs) and non-functional requirements (NFRs). The FPA method was expended and released in an internal IBM study [2].

In 1986,A nonprofit organization, which name is International Function Point User Group (IFPUG) promoting the PFA which can also for promoting and disseminating the software management ,software development and its maintenance. IFPUG's ISO/IEC 20926:2010 has specify FPA. ISO/IEC 20926:2010 standard define the definition, method ,rules, and steps for applying FPA. Several techniques have been derived from FPA and also standardized by ISO/IES like COSMIC, FiSMA, Mark-II, and NESMA. In our study, we mainly focus on the FPA [3]. FPA is used as input for computing the development effort, productivity, or cost of an information system or software. The importance of the FPA can be seen in the Brazil, where development of software procurement for the Federal Ruling party must be complying with some software metric like FPA [4].

In FPA, the user's perspective is employed for measuring the functionality of an information system, i.e., what a user get as an output based on their input. Computing the value of function point is a multi-stage execution, i.e., (a) unadjusted function point (UFP) (b) complexity adjustment factor (CAF). The function point measurement process is shown in Fig. 1.

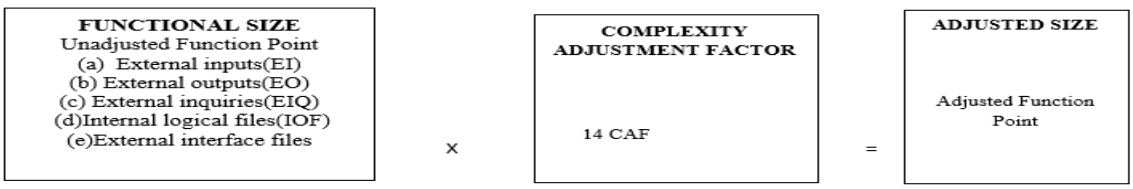

Fig. 1: A model for the measurement of function point

To compute the function point of software, the following five measuring parameters (MP) are identified from the system and requirements, i.e. (a) external inputs (EI), (b) external output (EO), (c) external inquiries (EQ), (d) internal logical files (ILF), and (e) external interface files (EIF); and then their counts are determined based on weighting factor, as shown in Table I.

MP	Weighting factor			Total
	Simple	Average	Complex	
EI =	3	4	6	
EO =	4	5	7	
EQ =	3	4	6	
ILF =	7	10	15	
EIF =	5	7	10	

Table I: Calculating function point

The following relationship is used for computing the function point (FP) of software:

FP = Total number of count $\times [0.65 + 0.01 \times \sum F_i]$ (1)

The F_i $(i = 1\ to\ 14)$ are the CAFs based on the results to the given below queries:

- **VAF 1:** After the designing of the system what are the system require backup and recovery in terms of reliability? (Two NFRs: Reliability and Recoverability)
- **VAF 2:** Inter system data communication/Intra system data communication is required?
- **VAF 3:** "The system is distributed executing functions/not distributed executing functions."?
- **VAF4:** The system performance is critical or not critical? *(*One NFR: Performance)
- **VAF 5:** Will the system is capable to execute in an existing, highly used operating environment?
- **VAF 6:** After designing the system, requiring on-line data entry?
- **VAF 7:** "Whenever applying the online data entry does the on-line data entry require the input transaction to be built over multiple screens or multiple operations?
- **VAF 8:** "The system is capable master files can be updated on-line or not?
- **VAF 9:** "Are the inputs of the system, output of the system, files of the system, or inquiries complex of the system"?
- **VAF 10:** "After the input to the system Is internal execution complex"?
- **VAF 11:** After designed the code, It should be reusable? (One NFR: Reusability)
- **VAF 12:** Are the system is usable after installation?
- **VAF 13:** Designed system is capable to install multiple times and multiple organizations?
- **VAF 14:** After designing the system, application can facilitate to modify in the system or easy to use by the stakeholders.

Based on the critical analysis of the above VAF, it is found that four NFRs have been included to calculate the FP value, i.e., Reliability, Recoverability, Performance, and Reusability. There are some NFRs which are also important during the development like security, maintainability, etc. These requirements are not the part of the VAF. In addition to this few authors have also identified the limitations in FPA methodology. For example, Kampstra and Verhoef [5] and Kemerer [6] reported that FPA method does not produce the results are consistent across numerous software measures. Therefore, to address the issues of VAF, this paper presents an extended list of VAF in which few NFRs have been added to compute the function point of software. The structure of this document is represents as: The related work on FPA is explained in Part II. The computation of function point of an information system is discussed in Part III. Finally, the result and future research are described in Parts IV.

II. Literature

The objective of this part is to discuss the related work in the domain of FPA. Various studies have been published based on the empirical analysis of FPA. For example, Abran *et al.* [7] focused on the empirical study of the measurement process of FPA. In their work, the authors found that there is a relationship between work-effort and FPA components. Most of the work on FPA has addressed the

issues related to its structure such as comparison with lines of codes, accuracy of estimation, productivity analysis, etc. Only several writers have reviewed FPA and its weaknesses. In [7] the empirical designs includes sets of "*independent variables*", "*empirical datasets*", "*identification of similar types of data set"*, and "*statistical analysis of the data variables*". Junior *et al.* [8] performed a comprehensive review of the literature on FPA to address the following research questions: (a) "Are there FPA problems being reported in terms of usability", (b) Improve the usability of the FPA's? Which improvements are being proposed?" (c)There is the recommended improvements is effective/usability/efficiency, (d) "What is the restrictions of the recommended growth". Issues in the area of FPA have been divided into three categories, i.e., "*weights and complexities*", "*adjusted functional size*", "*technological independence*". Among these, our work is focused on the adjusted functional size. Bharadwaj and Nair [9] focused that non-functional requirements like efficiency, usability, maintainability, and portability are not included in the VAF calculation. In another study, Abdullah et al. [10] pointed out that FPA does not cover how to compute the functional size of project based on information security area. Ahn *et al.* [11] also raised the associated problem to the VAF. Keeping in view the importance of the VAFs, in this work, we focused on analyzing the VAF according to the need of the project because the existing list of VAF is not sufficient for calculating the functional size of an information system or software; and it is discussed in the next section.

2.1. Computation Of Function Point Of An Information System

The aim of this section is to analyze the VAF for the requirements of an Institute examination System (IES). The goal of an Institute Examination System (IES) is to automate examination system task of an institute. Various studies have analyzed the requirements of an IES in the area of software requirements selection and prioritization. For example, Sadiq and Devi [12] analyzed the requirements of an IES using fuzzy soft-set theory. In another study, Nazim *et al.* [13] compared the fuzzy "*analytic hierarchy process*" and fuzzy "*technique for order of preference by similarity to ideal solution*" by considering the requirements of an IES. In this study, we have adopted the dataset of the requirements of an IES from the work of Sadiq and Devi [14]. A list of selected FRs of an IES is given below:

- FR1Creating the admit card for the end of semester examination
- FR2: An examination fee receipt is generated
- FR3: Online mode examination
- FR4: Generation of seating arrangement of an examination

To compute the function point value of the above selected requirements, first these requirements have been visualized and then analyzed based on the five measuring parameters and fourteen VAFs. The five measuring parameters of FR1, i.e., generation of admit card of end semester examination, have been identified after visualizing the FR1, see Fig. 2. The list of the measuring parameters is given below:

Number of inputs:

(Here, we have identified nine inputs)

i.Name of the Institute
ii.Logo of the institute
iii.Name of the Candidate
iv.Roll No
v.Session
vi.Examination Name (B. Tech. / M. Tech. / etc.)
vii.Upload Signature
viii.Upload Photo
ix.Examination Fee

Number of Outputs:

i. One output with following information, i.e., Date of examinations, Centre Name (for example, ABES Engineering College, Ghaziabad, India), and few set of instructions written on the back side of the hall ticket.

Number of inquiries:

i. There will be two inquiries, i.e., "Printout of the hall ticket" (Yes/No) and " Have you submitted the examination fee (Yes/No)"

Internal Logical File:

i. There will be one ILF in which data of all the students will be stored.

External Interface File:

i. There will be one EIF that will be related to the bank.

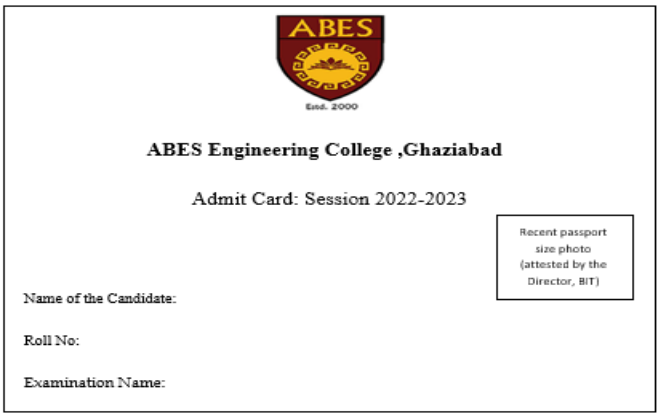

Fig. 2: Visual representation of FR1

The value of the five MPs for FR1 is given in Table II. Here, we assume that the weighting factors for FR1 as average. The same procedure was applied to calculate the count total of the remaining three FRs, i.e., FR2, FR3, FR4 of an IES and the results are summarized in Table III, IV, and V, respectively.

MPs	Count	Weighting factor	Total
EI	9	4	36
EO	1	5	5
EQ	2	4	8
ILF	1	10	10
EIF	1	7	7
Count Total			66

Table II: Value of the five MPs for FR1

MPs	Count	Weighting factor	Total
EI	9	4	36
EO	1	5	5
EQ	1	4	4
ILF	1	10	10
EIF	1	7	7
Count Total			62

Table III: Value of the five MPs for FR2

MPs	Count	Weighting factor	Total
EI	2	4	8
EO	1	5	5
EQ	1	4	4
ILF	1	10	10

EIF	1	7	7
Count Total			34

Table IV: Value of the five MPs for FR3

MPs	Count	Weighting factor	Total
EI	3	4	12
EO	1	5	5
EQ	2	4	8
ILF	0	10	0
EIF	0	7	0
Count Total			25

Table V: Value of the five MPs for FR4

After computing the MPs of FRs of an IES, the next step is to evaluate the given set of FRs based on the 14 VAFs. Based on our analysis, we found that the existing VAFs are not sufficient because it lacks various NFRs which are indispensible for the successful development of software. Thus, we have introduced the following VAFs in the list of existing VAFs:

- **VAF 15:** Security
- **VAF 16:** Cost
- **VAF 17:** Usability

Now the final list of the 16 VAF is shown in Table VI. The results after the evaluation of the given FRs based on the 17 VAF are also summarized in Table VI. Here we assume that weighting factors of all the VAFs are average.

VAFs(Value Adjustment	FR1	FR2	FR3	FR4
Value Adjustment Factor 1	3	3	3	3
Value Adjustment Factor 2	3	3	3	3
Value Adjustment Factor 3	3	3	3	3
Value Adjustment Factor4	3	3	3	3
Value Adjustment Factor 5	3	3	3	3
Value Adjustment Factor 6	3	3	3	3
Value Adjustment Factor 7	3	3	3	3
Value Adjustment Factor 8	3	3	3	3
Value Adjustment Factor 9	3	3	3	3
Value Adjustment Factor 10	3	3	3	3
Value Adjustment Factor 11	3	3	3	3
Value Adjustment Factor 12	3	3	3	3
Value Adjustment Factor 13	3	3	3	3
Value Adjustment Factor 14	3	3	3	3
Value Adjustment Factor 15	5	3	5	2
Value Adjustment Factor 16	5	3	5	2
Value Adjustment Factor 17	5	3	5	2

Table VI: Evaluation of the FRs of an IES based on VAFs

It has been observed that security varies from one requirement to another. For example, security is important for FR1, thus, it has the value 5 in Table VI. The cost of this requirements will also be high, thus, numeric value in Table VI is also 5. The same procedure is applied for evaluating the FRs based on the Value adjustment Factor (VAF), and the results are summarized in Table VI.

Based on our analysis, we found that existing fourteen VAFs are not sufficient for the development of secure and cost effective software or information system. Thus, these three NFRs should also be included in the list of VAFs.

III. Conclusion

This study concentrates on one of the major key points of the function point computation, i.e., value adjustment factor. In this paper, two VAFs have been introduced for analyzing the security and cost of software. The requirements of an IES have been evaluated based on the five measuring parameters and seventeen VAFs. After the computational process, we found that the total count of FR1, FR2, FR3, and FR4 are 66, 62, 34, and 25, respectively. These four requirements of an IES have also been evaluated based on the seventeen VAFs so that the cost of software can be computed by considering the security and usability of the requirements.

IV. Acknowledgment

This work has been carried out in collaboration with Indraprastha Research Laboratory, Indraprastha Institute of Information Science Private Limited, New Delhi, India (www.iiisc.in).

References

[1]Albercht, A.J.: “Measuring Appliction Development Productivity”, IBM Appl. Develop. Symp.,Montara, CA,USA,1079,pp.83-92
[2]IFPUG, “International Function Point Users Group”[Online], http://www.if[ug.org,2014
[3]IFPUG, “Function Point Counting Practices Manual” ,Release 4.3.1,International Function Point Users Group, Westerville ,OH, USA,2010.
[4]MPOG, “Instrucao Normative SLTI/MPOG (in Portuguese)”,Brazil’s Ministry of Planning, Budget and Management, Brasilia. Brazil,no.4,2010.
[5]Kampstra P., “Vrehoef,C.:Reliability of Function Points Counts”, UV University Amsterdam, Department of computer science, The Netherlands.
[6]kemerer C.F,” Reliability of Function Points Measurement”, A field Xeperiment, Commun , ACM , vol.36,no.2,pp.85-97,1993.
[7]Abran A, Robbilard P.N, “Function point analysis: An Empirical Study of its Measurement Processes”, IEEE Transactions on Software Engineering ,vol. 22, Dec,1996 .
[8]Junior M.D.F,; Fantinato,M.,Sun, “Improvements to the Function Point Analysis Method: A Systematic Literature Eeview”, IEEE Transactions on Engineering Managemen,2015.
[9]Bharadwaj A. ; Nair T.R , “Mapping General System Characteristics to Non- Functional Requirements”, IEEE International Advance Computing Conference, IACC 2009.
[10] Abdullah N.A.S, Abdullah R., Selamat M.H., Jaafar, A, “Software Security Characteristics for Function Point Analysis”, IEEE 16thInt. Conf. Ind. Eng.Manag. , pp. 394-397,2009.
[11] Ahn Y. , Suh J. ,Kim S. , Kim,H, “The Software Maintenance Project Effort Eastimation Model Based on Function Points,” Journal of Software.Maintenance, Evol., vol.15,no.2, pp.71-85,2003.
[12] Sadiq M., Susheela Devi V, “Fuzzy-Soft set Approach for Ranking the Functional Requirements of Software,”Expert Systems with Applications,vol. 193, May ,116452,2022.
[13] Nazim M., Mohammad C.W, Sadiq M., “A Comparison Between Fuzzy AHP and Fuzzy TOPSIS Methods to Software Requirements Selection,” Alexandria Engineering Journal, vol no. 61, Issue 12, pp 10851- 10870,2022.
[14] Sadiq M. , Devi V, “Prioritization and Selection of the Software Requirements using Rough-Set Theory,” IETE Journal of Research, pp.1-18,2021.