Experiment Instruction of Software Engineering Economics
Experiment-in-class 2
Cost Estimation of Software Project - Software Size Measurement

1. Experimental Purpose

Understand the principles of the function point method for software project size measurement and master the measurement process of the function point method.

Before the experiment, students need to preview the relevant knowledge of software function point analysis and measurement.

In the experiment, student-group of the course project as a unit and measures the software project size based on the design documents of course project in the course "Software Engineering Management and Economics". The function point analysis methods are recommended.

After the experimental, student-group should select a specific function point analysis method to measure the software size of the course project, and another function point analysis method should be used to verify the obtained software size data.

This experiment is a design experiment, with 1 hour of in-class experiment and 4 hours of out-of-class experiment.

2. Experimental Principles

Software size measurement is the basis for software engineering management, software cost estimation and software economic evaluation.

Software size measurement methods mainly include line of code method, function point analysis method, object point method and use case point method.

Function Point Analysis (FPA) is a method of measuring the size of software from the perspective of functional characteristics of the software system. Nowadays, the mainstream function point analysis methods include IFPUG, NESMA, GB/T 36964, COSMIC and MARK II. Among them, the market share of IFPUG or NESMA method exceeds 90%. GB/T 36964 method and the NESMA detailed function point method are basically equivalent to the IFPUG method.

3. Experimental Steps

This experimental guide recommends the NESMA method or the IFPUG method.

It is recommended that you follow the steps below to perform the software sizing process.

3.1 Determine the engineering type of software system

Function point measurement of software size can include the entire process of the software life cycle, from software requirements, architecture design, component design, testing to software deployment and maintenance. The accuracy of function point measurement of software system size is gradually improving. There are three project types for function point measurement software systems: new development projects, enhancement (development) projects, and application function point measurement.

New development projects: Function point measurement of new development projects refers to the functions provided to users when the software is first installed and measured. These functions include both function points contained in the application itself and function points generated by data conversion. The measurement of function points of new development projects must be updated as the development of the software project progresses. Subsequent measurements do not have to be done from scratch, but they must verify the functionality that has been identified and try to capture new functionality, often referred to as "scope extension."

Enhancement (development) project: Enhancement (development) project function point measurement refers to the modification of the existing software system and the combination of functions brought to the user due to the addition, deletion and change of functions. When implementing deletion functionality in an enhancement (development) project, the workload may increase exponentially even though the size of the software is reduced. Enhancement (development) project function point measurements may also include function points resulting from data transformation.

Application function points measure an installed application. This is also called a baseline calculation or installed calculation, and evaluates the capabilities your application currently provides to users. The Total Installed Application Function Points measure for a task represents the sum of all application function points that are currently being used and maintained.

3.2 Identify and determine software system boundaries and scope

In function point measurement, boundary delineation and determination are the key and difficult points in correctly estimating the size of the software system being measured. The measurement scope is determined based on the purpose of the measurement work. It is generally believed that the measurement boundary refers to the interface or boundary between the program under test and external users, other application systems or programs.

3.3 Function Point Analysis

Function point analysis divides the functional requirements of the software system into data functional requirements and transaction functional requirements for processing data. Data functional requirements are described by data type function points, which include the internal logical data of the application and the external interface data of the application. Transaction functional requirements are described by function points that reflect the type of human-computer interaction (transaction processing), including external input of data, Output and query.

3.4 Measurement data function points and transaction function points

3.4.1 Data type function points

- ILF Internal Logical File (Internal Logical File)
- EIF External Interface File (External Interface File)

The ILF internal logical file is a set of logically associated data or control information within the user-identifiable system boundary. The ILF is maintained by the basic processing process of the system.

ILF identification rules: The following conditions must be met

- Collections of data and control information are logical and user-identifiable.
- Data collections are maintained within the boundaries of the system by basic processes.

The EIF external interface file is user-identifiable and maintained by other systems. It is a set of logically related data or control information referenced in this system. It is referenced by the basic processing process of this system and is an internal logical file of other systems.

EIF identification rules: The following conditions must be met

- Collections of data and control information are logical and user-identifiable.
- The data group is referenced from outside the system.
- The data group is not maintained by the system.
- The data group is maintained as an ILF by another system.

The calculation of the complexity of ILF and EIF can be simply understood as the calculation of the complexity of business data. The complexity is determined by the data element type (DET) and record element type (RET).

3.4.2 Function points of human-computer interaction type (transaction)

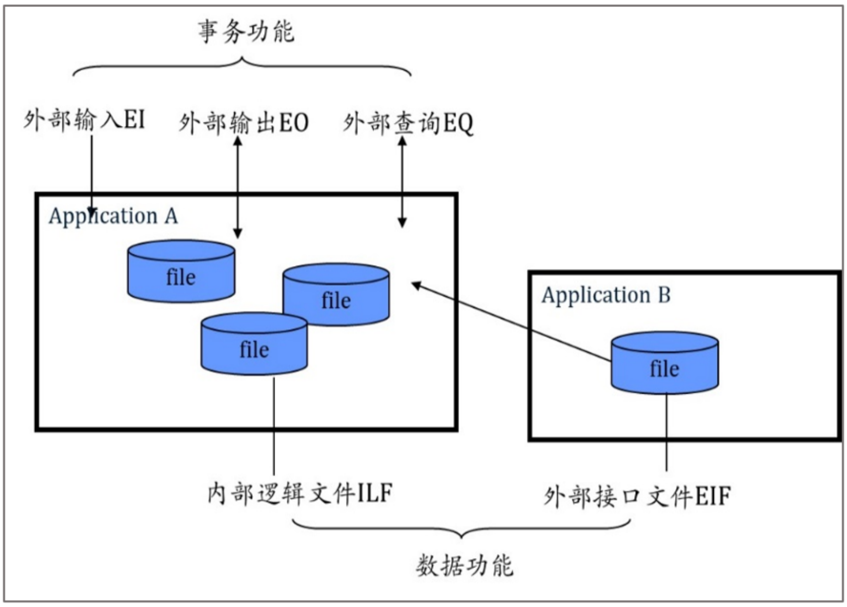
- EI External Input (External Input)
- EO external output (External Output)
- EQ External Inquiry

EI external input refers to a basic processing process that processes a set of data or control information from outside the boundaries of this application. The basic purpose of external input is to maintain (including adding, modifying and deleting data, etc.) an internal logical file (ILF) or to change the behavior of the system.

EO external output refers to a basic processing process that provides processed data or control information to outside the application boundary or to users. The basic purpose of external output is to display to the user a set of data or control information that has been processed by other logic besides extraction. Other processing here includes at least one mathematical operation or generation of derived data. EO external output can be in two ways, data provided to external applications and data displayed to users. A typical output is the various reports generated by the system.

EQ external query refers to a basic process of sending data or control information outside the application boundary. The basic purpose of external queries is to display extracted data or control information to the user. The logical processing of external queries does not contain mathematical formulas or calculations, does not modify the ILF, and does not change system behavior.

The calculation of the complexity of EI\EO\EQ can be understood as the calculation of the complexity of business implementation. The complexity is determined by the data element type (RET) and the number of referenced file types (FTR).



3.5 Calculating Unadjusted Function Points

According to the requirements model and design model of the software system, count the number of ILF, EIF, EI, EO and EQ components and determine the complexity level of each component. According to the standard calculation method of the International Function Point User Group (IFFPUG method), the unadjusted function points (UFP) of the software system are calculated with the following formula:

$$\text{UFP} = \sum ILF + \sum EIF + \sum EI + \sum EO + \sum EQ$$

3.6 Calculate adjusted function points

In order to effectively reflect the impact of non-functional factors of software systems on software project workload, the IFPUG method uses 14 general system characteristic factors to correct the unadjusted function points. These characteristic factors include data communication, distributed data processing, performance, complex processing and reusability etc.

These 14 factors are assigned a weight value from 0 to 5 according to their different degrees of impact on the software system. The function points of the application system are adjusted according to the following formula, and finally the function points of the software system workload are obtained.

$$\text{FP} = \text{UFP x VAF}$$

In the above formula, UFP is the unadjusted function point and VAF is the function point adjustment factor. VAF calculation formula:

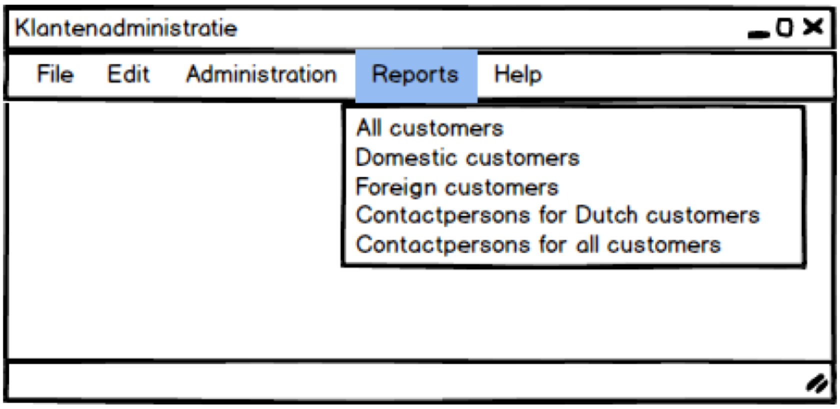
$$\text{VAF} = 0.65 + 0.01\text{x} \sum_{1}^{14} \text{Ai}$$

In the above formula, Ai ranges from 0 to 5, so the value range of VAF is from 0.65 to 1.35.

4. Case Study: Software Size of Customer Application

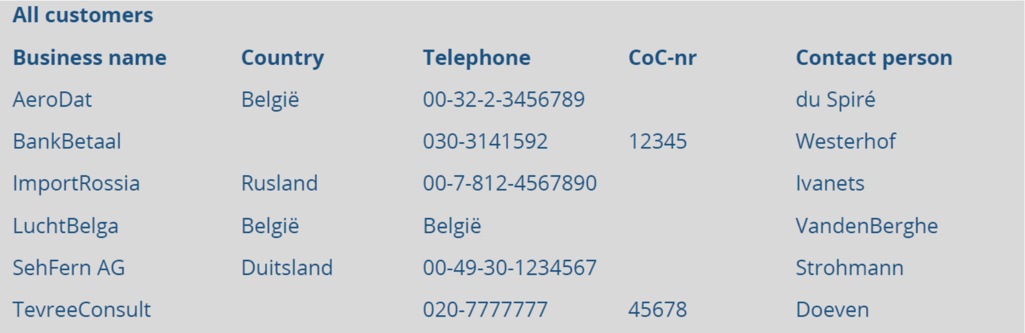
Problem description

The functional specifications below have been made for a small customer application at an early stage of application development.
The following is maintained for each customer: Name, Address, City, Country Code, Telephone, and Contact Person. The registration numbers of Dutch customers registered at the chamber of commerce (CoC) are also maintained. Users would like to be able to add, change, and delete data. When a user wants to change and delete data, the customer data present must be shown for verification. Users also want to be able to print the following reports via the menu below:



A sketch of each of these reports is given on the following page. The name of a country is retrieved from a file called Countries that contains the name of a country for each country code. This file is maintained by a different application.

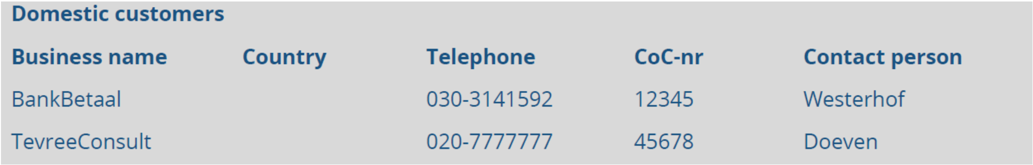
1. Report "All customers"
This report contains all customers and is ordered by company. The country for Dutch customers is not printed.



All customers

Business name	Country	Telephone	CoC-nr	Contact person
AeroDat	België	00-32-2-3456789		du Spiré
BankBetaal		030-3141592	12345	Westerhof
ImportRossia	Rusland	00-7-812-4567890		Ivanets
LuchtBelga	België	België		VandenBerghe
SehFern AG	Duitsland	00-49-30-1234567		Strohmann
TevreeConsult		020-7777777	45678	Doeven

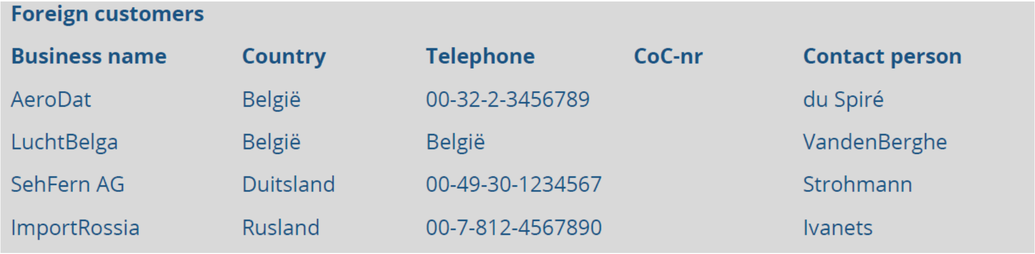
2. Report "Domestic customers"
This report contains all Dutch customers ordered by company.



Domestic customers

Business name	Country	Telephone	CoC-nr	Contact person
BankBetaal		030-3141592	12345	Westerhof
TevreeConsult		020-7777777	45678	Doeven

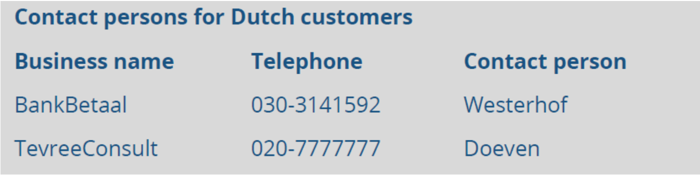
3. Report "Foreign customers"
This report contains all foreign customers. The user wants the country to appear at the beginning of each line on the list.



Foreign customers

Business name	Country	Telephone	CoC-nr	Contact person
AeroDat	België	00-32-2-3456789		du Spiré
LuchtBelga	België	België		VandenBerghe
SehFern AG	Duitsland	00-49-30-1234567		Strohmann
ImportRossia	Rusland	00-7-812-4567890		Ivanets

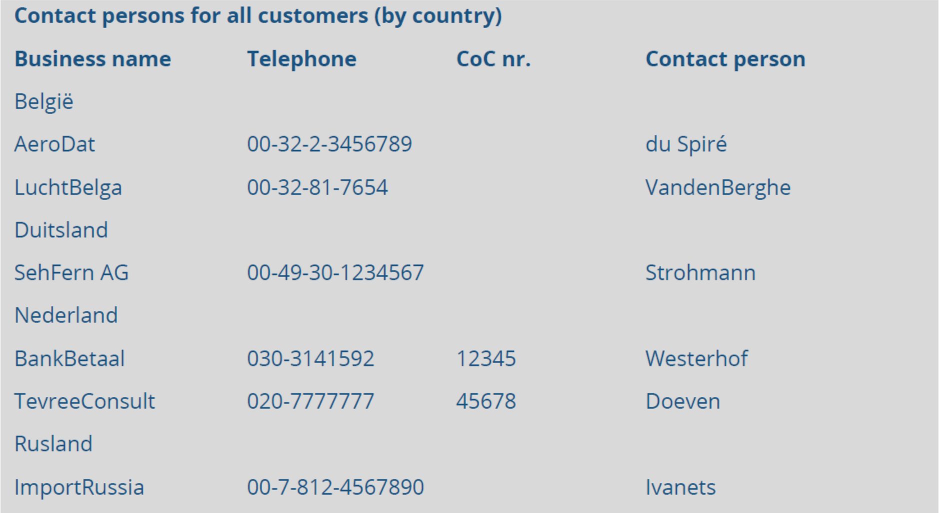
4. Report "Contact persons for Dutch customers"
The report contains the telephone number and the contact person of all Dutch customers.



Contact persons for Dutch customers

Business name	Telephone	Contact person
BankBetaal	030-3141592	Westerhof
TevreeConsult	020-7777777	Doeven

5. Report "Contact persons for all customers" (by country)
This report contains the telephone number, the chamber of commerce number, and the contact person of all customers. Customers are grouped by country.



Contact persons for all customers (by country)

Business name	Telephone	CoC nr.	Contact person
België			
AeroDat	00-32-2-3456789		du Spiré
LuchtBelga	00-32-81-7654		VandenBerghe
Duitsland			
SehFern AG	00-49-30-1234567		Strohmann
Nederland			
BankBetaal	030-3141592	12345	Westerhof
TevreeConsult	020-7777777	45678	Doeven
Rusland			
ImportRussia	00-7-812-4567890		Ivanets

For this application a high level function point analysis has to be carried out.

Discussion

The entity type Customer can be maintained in the application and is an internal logical file. Country is an FPA table that the application can only read. This is counted as a record type in the FPA tables ELF. Other FPA tables do not exist; therefore, the FPA tables ELF in this case consists of only one record type.

The specifications indicate that customer data can be added, changed, and deleted.

This means that three external inputs are identified. The fact that a chamber of commerce number may not be entered for foreign customers does not play a role.

The user has not requested a separate external inquiry. The showing of current customer data for the purpose of verification when a user changes and deletes data is not counted as a separate external inquiry.

Reports 1, 2, and 3 together count as one external output because the following applies in all cases:

- The same object is being reported on (customer)
- The selection criterion is the same (country)
- The processing in order to produce the output products is the same (Except for the selection mechanism, no additional processing is needed.)
- The logical layout of the output products (set of data element types and their structure) is the same; i.e., business name + (country) + telephone + (CoC-nr) + contact person. The parentheses denote optionality. The sequence is not important.

It is irrelevant that a heading is not printed in all cases, as when data is not present or desired; e.g., a CoC-nr or the name of a country, respectively. The headings, after all, have been defined for the output product. Although the sequence of the columns is different in report 3, this is no reason to identify a separate external output. In these three cases, a direct selection takes place via the heading Country.

The same result could also be realized with a fill-in screen in which the user is provided with country code as a selection criterion. The fill-in screen would not be counted as a separate external input. Within FPA, the data to be filled in would be considered control information for the external output, and each piece of data would be included in the analysis as a data element type.

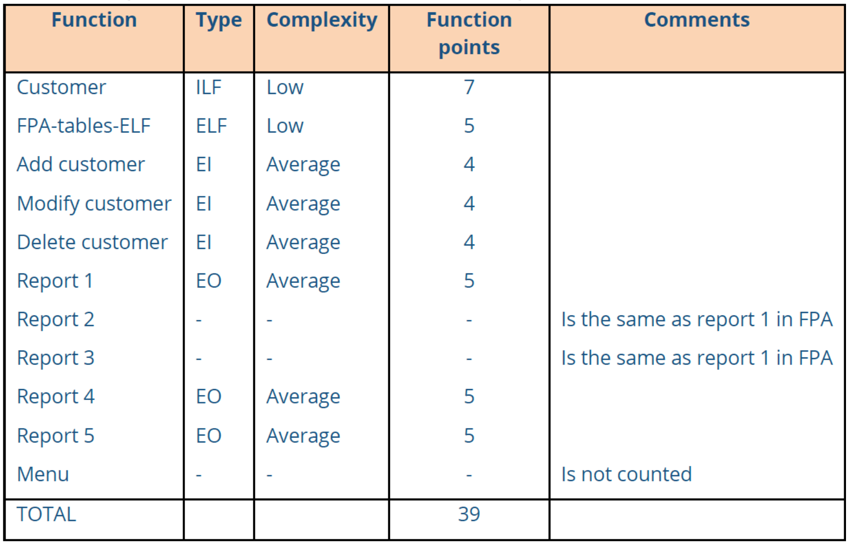
While it is true that report 4 selects the same customers as report 2, the logical layout is different because the set of data element types in report 4 is different: business name + telephone + contact person. Report 4 therefore counts as a separate external output.

Report 5 selects the same customers as report 3. The set of data element types is the same in both reports. However, the structure of the output product is different (the data element types are grouped differently) because the country is presented once each time. Therefore the logical layout is different. As a result, report 5 is identified as a separate external output.

According to the guidelines, no transactional functions are identified at all for the FPA tables ELF, even if external inquiries or external outputs would be present.

Solution

A logical file is counted as low in a high level function point analysis and a transaction as average. This results in the following functional size:



Function	Type	Complexity	Function points	Comments
Customer	ILF	Low	7	
FPA-tables-ELF	ELF	Low	5	
Add customer	EI	Average	4	
Modify customer	EI	Average	4	
Delete customer	EI	Average	4	
Report 1	EO	Average	5	
Report 2	-	-	-	Is the same as report 1 in FPA
Report 3	-	-	-	Is the same as report 1 in FPA
Report 4	EO	Average	5	
Report 5	EO	Average	5	
Menu	-	-	-	Is not counted
TOTAL			39	

The unadjusted functional size of the application is 39 function points.

5. Content of experiment report and Deadline Time

Students must submit a hand-written experiment report (PDF format) on Canvas. This experiment report must include the following 4 essential contents:

- Experimental background.
- Experimental content and steps.
- Experimental results (Size of your course project).
- Your learning experiences in this Experiment.

The time scale to finish experiment report is Three weeks. The deadline for submitting the experiment report is before 23:59 on June 1. Each student should upload the experiment report on Canvas. Students who submit late will be penalized points.

6. Thinking

Verify experimental results.

It is recommended to use the NESMA standard measurement model and method to measure the workload of this experimental case. Are there any differences between the NESMA standard and the IFPUG standard? If so, what's the difference?

7. Reading

ISO/IEC 20926:2009 Software and systems engineering software measurement IFPUG functional size measurement method.

ISO/IEC 24570:2018 Software Engineering NESMA Functional Size Measurement Method.

END