ISSN: 0798-1015

REVISTA ESPACIOS

Vol. 41 (32) 2020 • Art. 6

Recibido/Received: 24/06/2020 • **Aprobado**/Approved: 27/07/2020 • **Publicado**/Published: 27/08/2020

Agile counting process of software product maintenance size: A statistical analysis

Processo de contagem de tamanho ágil de manutenção de produto de software : Uma análise estatística

CALAZANS, Angélica T.S.[1]
MARTINS, Elcio G.P. J.[2]
MASSON, Eloisa T.S.[3]
TEIXEIRA, Roberto G.[4]

Abstract
The objective was to analyze the customization of a functional size metric with a focus on reducing the effort of the process of counting maintenance projects, experimenting and validating. For this, a bibliographical research was carried out to identify the existing proposals in the literature regarding the detailed Function Point Analysis (FPA) and estimated NESMA counts. Aspects related to the counting effort with the FPA and Nesma were also identified. 13,320 detailed maintenance counts (FPA) of a Brazilian organization were analyzed. In 68%, estimated count was lower than detailed. Transactional functions were the ones that most impacted the differences between estimated and detailed. So, was applied to the proposed customization, called Agile Counting. The Agile Count (NESMA customization) was applied in counts with a size above 51 FP (2915 counts) and the result was promising (variation lower than 1.21% in relation to the detailed count). The work done shows that one can still analyze and identify improvement opportunities in well-founded models (FPA and NESMA), thus increasing the applicability and minimizing the disadvantages of these models.
Key words: maintenance, FPA, NESMA, software size measurement effort

Resumo
O objetivo foi analisar a customização de uma métrica funcional de tamanho com foco na redução do esforço do processo de contagem de projetos de manutenção, experimentar e validar. Realizou-se pesquisa bibliográfica, e identificou-se na literatura as propostas existentes que analisavam a Análise de Pontos de Função (APF) e a contagem estimativa NESMA. Aspectos relacionados ao esforço de contagem utilizado a APF e NESMA foram identificados. Foram analisadas 13.320 contagens de manutenção detalhadas (APF) de uma organização brasileira. Em 68%, a contagem estimada era menor que a detalhada. As funções transacionais foram as que mais impactaram as diferenças entre estimadas e detalhadas. Foi aplicada a Contagem Ágil (customização da NESMA) nas contagens com tamanho acima de 51 PF (2915 contagens) e o resultado foi promissor (variação inferior a 1,21% com relação a contagem detalhada). A pesquisa mostra que ainda podem ser analisados e propostas melhorias em modelos conhecidos e estáveis (APF e NESMA), de forma a melhorar a aplicabilidade e minimizar as desvantagens desses modelos.
Palavras chave: manutenção, FPA, NESMA, esforço de mensuração de manutenção de software

[1] Professor. Centro Universitário de Brasília – Uniceub, Brasília- DF, Brazil. Email: angelica.toffano@gmail.com
[2] Consultor. Caixa Econômica Federal. Brasília – DF, Brazil. Email: elcio.martins@caixa.gov.br
[3] Professor. Centro Universitário de Brasília – Uniceub, Brasília- DF, Brasil. Email: eloisa.masson@caixa.gov.br
[4] Consultor. Caixa Econômica Federal. Brasília – DF, Brazil. Email: Roberto.teixeira@caixa.gov.br

1. Introduction

One of the challenges of software engineering is the reliable sizing of software, for new development and, also for maintenance. Several surveys indicate that software maintenance consumes 60% to 80% of the total of lifecycle cost. These surveys also report that maintenance costs are largely due to improvements, generally 75% - 80% (Calazans, Martins, Masson, & Teixeira, 2017). Despite these numbers, the area of software maintenance estimation has received less attention in relation to the new development.

Several approaches to software size measurement have been developed and are being applied at different stages of the development lifecycle in the last three decades. Maintenance size, or software size, can be defined considering the work products (number of programs, lines of code, objects, etc), or can be defined in relation to the functionality delivered to the client (input transactions, reports etc.), among other possibilities. All these approaches have advantages and disadvantages.

Functional metrics have the advantage of being applicable early in the software lifecycle. The need for accurate size estimates and project predictions is one of the important issues in the software industry. The importance of measuring software size is recognized by models such as CMMI (SEI, 2010) and MPS-BR (Softex, 2016). Thus, approaches to measure the size of new developments and maintenance based on the software functional size have been well studied and some functional models are already recognized as standards (ISO/IEC, 2002), (ISO/IEC, 2009), (ISO/IEC, 2011), (ISO/IEC, 2005).

Functional size metrics are used to measure the software from the perspective of the final users and considering the amount of functionality to be delivered. These measures can be used for project estimates, quality assessment, benchmarking, productivity management, contract outsourcing management, among other aspects (Vazquez, Simoes, & Albert, 2013).

For (Raju & Krishnegowda, 2013), some of the disadvantages of these metrics are: requiring people with experience to perform this activity and a process of reviewing those counts; the counting process consumes time, effort and costs (Lavazza, 2015) and the counting process is based on documentation, which must contain certain requirements to be performed correctly and consistently.

The most referenced functional metrics, according to (Calazans, Paldes, & Mariano, 2015), are COSMIC - Common Software Measurement International Consortium and FPA - Function Point Analysis. It is interesting to note that FPA adaptations, such as the NESMA proposals, are also referenced (Morrow, Wilkie, & McChesney, 2014), (Wang, Li, & Yu, 2008). In the view of (Garmus, 2001) the FPA is one of the methods most used by industry.

According to (Jones, 2013) FPA can be considered a universal software metric. The Brazilian government indicates the use of FPA in all of its software contracts. In addition, South Korea and Italy governments can follow this trend soon (Jones, 2013).

In Brazil, most of the public institutions that use FPA use Nesma's estimated proposal at beginning of the development or maintenance process, and later, when all the requirements are known and well defined, apply the FPA proposal to ratify or rectify the estimated initial count. Some examples of this working method: (Brazil, 2014), (Badesul, 2015), (Brazil, 2014), (Brazil, 2015), (Brazil, 2012).

Considering advantages and disadvantages described, and although the functional metrics, specifically the FPA and Nesma's estimation proposal, can be considered well-founded models, opportunities for improvements in these models can be analyzed and identified through comparative case studies, thus extending the applicability and minimizing the disadvantages of these models.

Thus, the present research aims to answer the following questions: is it possible to adapt a functional measurement model for maintenance projects in order to reduce time, cost and effort of the counting process? Will the results obtained in terms of measurements be compatible with the original metric used?

This work relevance is to contribute to improve a functional size measurement metric with a focus on maintenance, in order to reduce the effort and cost of counting. For both industry and academia, this proposal and evaluation are important. For the industry as a way to reduce costs and efforts from counting to maintenance, and to academia in the systematic search for alternatives to estimate size of software product maintenance.

In order to answer the proposed questions, the research objective is to customize a software-size functional metric focused on reducing time, effort and cost of the maintenance project counting process and to test and validate this new industry counts. For this, section 2 details the proposed metrics related to this subject. Section 3 explains the methodology used in the research and the proposed approach, while section 4 presents results and analysis. In Section 5, the research findings are synthesized.

2. Proposed metrics

2.1. Function Point Analysis (FPA), advantages and disadvantages

Function Point Analysis (FPA) measures software size by quantifying its functionality, based on the logical design or data model according to the end user vision and requirements. Currently FPA is recognized as ISO/IEC 20926 standard (ISO/IEC, 2009).

FPA main characteristics are: to be technology independent, to be applicable from the beginning of the system, to support the productivity and quality analysis and to estimate software size with a standard unit of measure.

FPA considers the data functions, divided into Internal Logical Files (ILF - which are logical groups of data held within the application boundary) and External Interface Files (EIF - files referenced by the application) and the transactional functions, divided into External Inputs (EI), External Outputs (EO) and External Inquirys (EQ).

Each data or transactional function will have a different weight depending on its complexity. Several tables based on the amount of data elements, records, and referenced files are used to determine the complexity of each function in Low, Medium, or High. Table 1 presents the FP quantitative by complexity of each data and transaction function and the percentage of variation among complexities.

Table 1
FP Number by function type and complexity and percentage
of size variation considering Medium complexity –

Function Type	Low (L)	Variation ((M-L)/M)	Medium (M)	Variation ((M-H)/H)	High (H)
ILF	7 FP	42%	10 FP	-33%	15 FP
EIF	5 FP	40%	7 FP	-30%	10 FP
EI	3 FP	33%	4 FP	-33%	6 FP
EO	4 FP	25%	5 FP	-28%	7 FP
EQ	3 FP	33%	4 FP	-33%	6 FP

Adapted Source (Herron, 2006)

The result of counting data and transactional functions is a measure called unadjusted function points (NoFP not adjusted) because it does not consider details that affect the product and its construction. The measurement adjustment is done through the Adjustment Factor, set of 14 characteristics that will influence the complexity of the software.

According to Jones (2013), the metric has strengths such as: FPA have projects measured more than all other metrics combined and hundreds of FPA counters certified by IFPUG are available in most countries. Also defined as the standard of most parametric estimation tools, such as KnowledgePlan, SEER, and Software Risk Master. In addition, (Heeringen, 2015) mentions some advantages of the functional metrics that are: the size measurement is carried out in an objective way, that is, two certified counters obtain approximately the same size, considering the same software product which makes the measure repeatable and verifiable.

The weaknesses of the FPA would be the slow process of counting using the FPA. Counting velocities for function points take on average 500 function points per day (Jones, 2013). Already (Herron, 2006) presents Table 2 with average time in hours to perform the detailed counts.

Table 2
Average productivity in hours for detailed
FPA counting, considering size in FP

Size	Effort in hours for detailed FPA
<50 FP	2,5
50 a 150 FP	4,3
150 a 300 FP	8,8
300 a 650 FP	13,9
650 a 1000 FP	20,8

Source (Herron, 2006)

However, the study of (Morris, 2004) shows that average productivities are worse as more detailed counting techniques are applied, according to Table 3.

Table 3
Productivity for counting considering
the level of the count

Counting level	Average Productivity
Interconnected and annotated count	200 FP/ day
Detailed and annotated count	250 FP/ day
Detailed count	300 FP/ day
Standard complexity count	400 FP/ day
Non-detailed count	750 FP/ day
Approximate size	Most applications can be estimated in half a day

Source (Morris, 2004)

For (Jones, 2013), due to the slow speed of the function point analysis, function points are almost never used in large systems, with more than 10,000 function points in size. This factor makes the FPA measurement process costly. Assuming a daily counting speed of 500 function points and a daily consulting fee of $ 1,500, counting an application of 10,000 function points, it would require 20 days and would cost $ 30,000. It could consider a cost of $ 3.00 for each counted function point.

Considering other points to improve in FPA, (Freitas Junior, Fantinato, & Sun, 2015) made a systematic review on FPA. The improvements proposed in the 18 selected FPA-related studies were analyzed and grouped into three categories: 1) "weights and complexities" determined for each data and transaction function; 2) "technological independence" and 3) calculate the "Functional size adjustment".

2.2. NESMA Approach

The Netherlands Software Metrics Users Association (NESMA) is an association of metric users that has proposed counting alternatives using the FPA in order to be able to measure a software product at the beginning of the process even though it does not have all the information about the functions data, transactions and general system characteristics (ISO / IEC, 2005). NESMA proposes Indicative and Estimated counts, in addition to having a complete proposal. The IFPUG itself suggests using Indicative and Estimated approaches to get the account at the beginning of the process (Timp, 2015).

The Estimated Count, proposed by NESMA (Vazquez, Simoes, & Albert, 2013), makes possible the size estimation from the identification of all the functionalities of the software. Using the complexity classification of the IFPUG, it applies the low complexity for each data function (ILF - 7 FP and EIF - 5 FP), and the average complexity for each transaction function (EI - 4 FP, EO - 5 FP and EC - 4 FP).

This approach does not address the application of the 14 General Characteristics of System to obtain the adjustment factor, that is, the Adjustment Factor receives a value of 1.

2.3. Other proposals

There are many proposals that try to mitigate the negative factors of size counting for development and maintenance, such as the counting effort, or even the time spent for this activity. The following are some related to the context of this article (FPA and Nesma's Approach).

(Meli, 2011) proposed a new approach called Simple Function Point. This approach would be an alternative to Function Point to quickly estimate the size of a software product at the beginning of the measurement process when all the information necessary for the detailed FPA count is not available. In this proposal new concepts are created: UGDG and UGEP. In this approach the application size in function points is given by the total number of data functions (UGDG) x 7 FP and by the total number of transaction functions (UGEP) x 4.6 FP (SiFPA, 2014) . (Ferrucci, Gravino, & Lavazza, 2016) applied this proposal in 25 WEB applications of a software company, obtaining promising results, but with low quantitative of experimentation.

(Herron & Dennis, 2011) presents a proposal called FPA Lite. With the same objective, this proposal suits Nesma's estimated proposal. After studies in two data groups with 30 and 95 maintenance projects, FPA Lite suits the complexity of data functions, from low to medium. The results presented with these data are promising. (Matsutani & Ribeiro, 2015) applied FPA Lite to 152 projects of the organization, and the results obtained were positive considering the low number of projects.

(Jones, 2013) suggests to mitigate negative factors, including counting effort, a high-speed scaling method that is embedded in Software Risk Master ™ (SRM) by scaling and estimating the tool under development by Namcook Analytics LLC. The sizing method provides application size considering a total of 15 metrics, including FPA, SNAP for non-functional attributes; COSMIC function points, Story points, User case points, code statements, and more. The author's proposal is a software for counting, which according to the author, would speed up and make the counting process more effective.

(Calazans, Martins, Masson, & Teixeira, 2017) suggests an adaptation of the Nesma count in order to streamline the counting process without impacting the results of the detailed count. They applied the proposal on 10,405 maintenance counts below 50 FP and the results were promising. The method proposes to use the Nesma count with a small adaptation. Consider the transactional functions EI with the high complexity, i.e. considering 6 FP. The present work analyzes this proposal for the other maintenance sizes.

3. Methodology

This work aims to analyze the customization of a software size functional metric (called Agile count) focused on reducing the time, effort and cost of the maintenance project counting process and to test and validate this approach in industry counts. Specific objectives are:

- Identify the software size counting process used by some Brazilian public organizations;
- Study existing customization proposals to reduce the time, effort and cost of counting development and maintenance projects;
- Evaluate the customization of proposed metrics, called Agile count (Calazans, Martins, Masson, & Teixeira, 2017) and, if feasible, use it in the analysis of the available data;
- Apply and compare the customized approach with the other proposals found.

The type of research used is classified as applied and empirical research, since it seeks to solve a concrete problem, which is the reduction of time, effort and cost of the counting process of maintenance projects. And empirical, because it identifies the correlation between the proposals found and the customized proposal.

In relation to the investigation means, we used: bibliographic research, documentary research, supported by documental research of the systems and projects that were measured.

Considering the phases of the cycle (ISO / IEC 15939, 2007), the steps for the analysis of these data and the proposal of a customized metric were defined. The process (ISO / IEC 15939, 2007) consists of four iterative activities:

- Establish and sustain the measurement process - identification of the problem, information needs, objectives etc;
- Plan the measurement process - identification of: the scope of the analysis, the organization, the counting process, the data to be used; definition of how the analysis, evaluation and what tools will be used, among other activities.
- Perform the measurement process - perform the data collection, analysis, identification of differences, standards and customization proposal; Analysis and evaluation of results.
- Evaluate the measurement - If necessary, adapt the customization proposal.

To meet the first activity (a), were analyzed by means of documentary research: the software counting process used by some Brazilian public organizations, the existing proposals that aim to reduce the time, effort and cost of the counting process and its applicability in academy projects or industry.

For the 2nd and 3rd activities (b and c), the organization and its characteristics were identified. The research used data from a large Brazilian public organization that operates in the financial market. In the process of counting maintenance projects, the organization performs at least two counts, one at the beginning of the process (estimated NESMA) and another, when all the necessary information for the detailed count (IFPUG) is already available. The organization does not apply the general system characteristics in any of the counts.

The data collection period of the counts was performed between 10/17/2016 and 10/21/2016. All counts had detailed count data (FPA IFPUG). For this work was calculated, based on each detailed count, the estimated FP quantitative. (NESMA). The Minitab tool was used for data statistical analysis.

Initially, 15,101 project counts were analyzed between improvement (maintenance), application and new development. The organization works with several development methodologies (RUP, Agile Methods, Structured, etc.) and the projects considered covered these methodologies. The organization in question has a high level of outsourcing software development. FPA size counts, both estimated and detailed, are also outsourced to a metric factory.

4. Results and analysis

Initially, the quantitative of projects considering the classifications was identified: maintenance, application and new development. Table 4 shows this distribution in the studied sample of 15,101 counts.

Table 4
Quantitative counts by classification

Count Type	Number of counts	Percentage	FP Qty – detailed count	FP Qty – estimated count	Percent difference
Application	1381	9,15%	374.691 FP	339.819 FP	9,30%
Improvement or maintenance	13320	88,21%	717.242 FP	558.043 FP	22,19%
New development	400	2,65%	98.544 FP	87.913 FP	10,78%
Total	15101	100	1.190.477	985.775 FP	17,19%

Source (Calazans, Martins, Masson, & Teixeira, 2017)

As can be observed, the initial analysis of the detailed counts presented an asymmetrical distribution. Thus, we opted to work initially, with the demands of improvement or maintenance projects. Of the 13320 project improvement or maintenance counts, 10,405 counts were for projects with less than 50 FP. The work of (Calazans, Martins, Masson, & Teixeira, 2017) analyzed and proposed an adaptation with promising results for counting range below 50FP (inclusive). It was decided, in this work, to consider all other maintenance counts above 51 FP (2915 counts) as the scope of the analysis to be carried out (Planning Step). Table 5 shows the ranges defined for counts above 51 FP.

Table 5
Size ranges analyzed in the research

Size Ranges	Number of Counts	Percentage	FP Qty – estimated count	FP Qty – detailed count	Percent difference
51 to 200 FP	2217	76,05	174244	213108	18,23
201 to 500 FP	493	16,91	124581	150890	17,50
501 to 4474 FP	205	7,03	158455	195082	18,77
Total	2915		457280	559080	

Figures 1, 2, and 3 presents the descriptive statistics of these new samples, which presented a little more symmetry.

Figure 1
Descriptive analysis of the sample counts from 51 to 200 FP

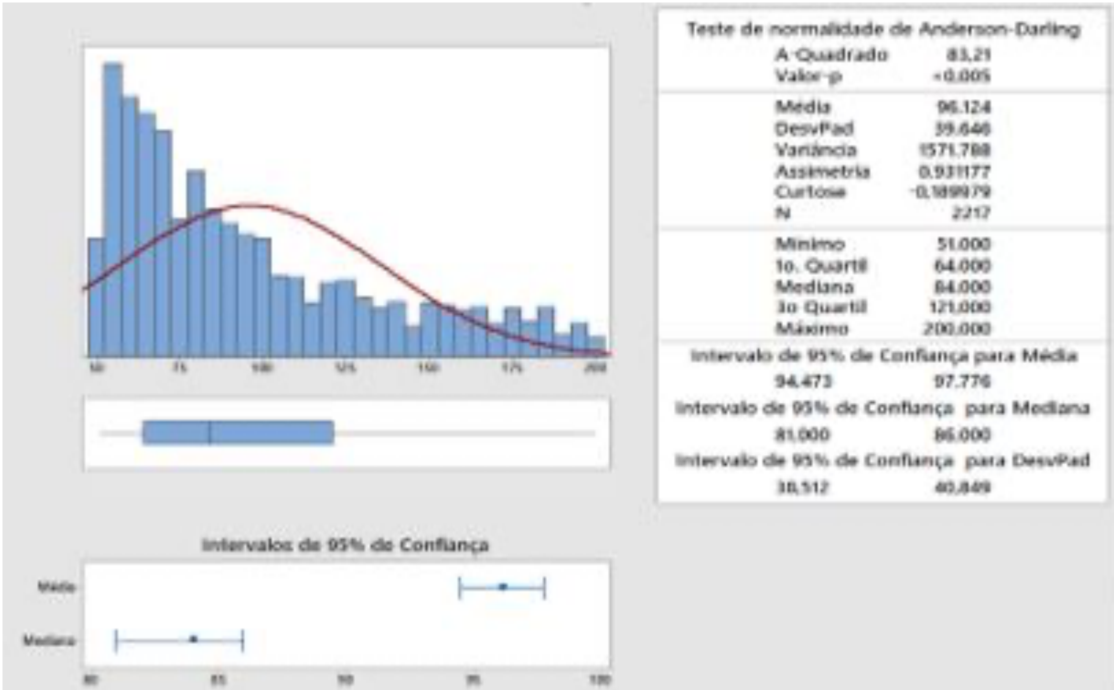


Figure 2
Descriptive analysis of the sample counts from 201 to 500 FP

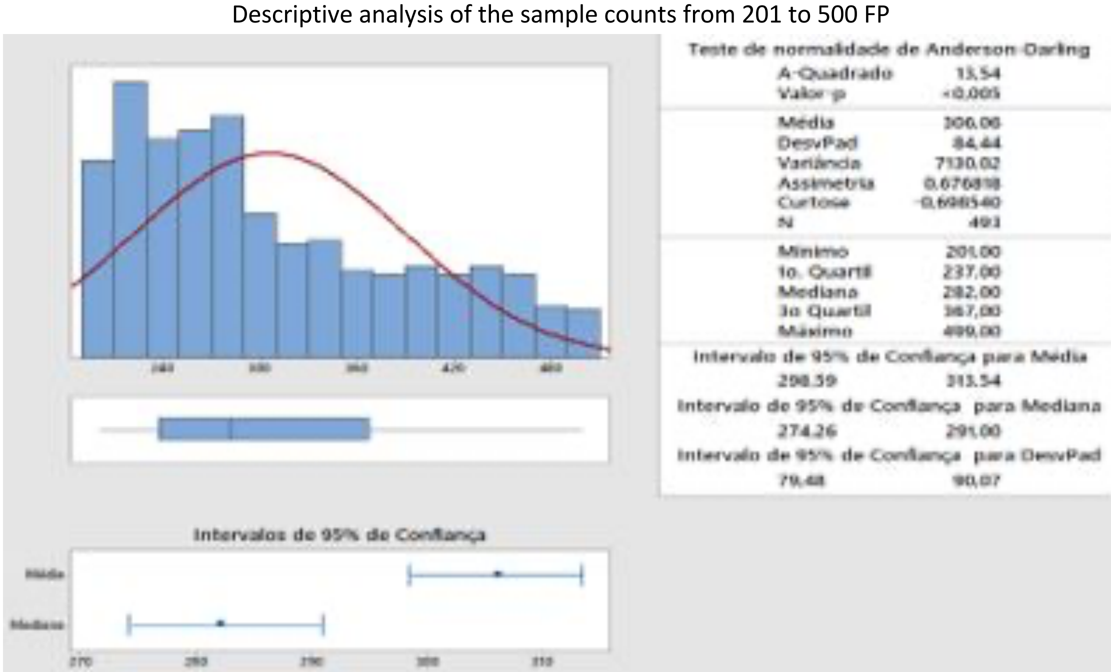
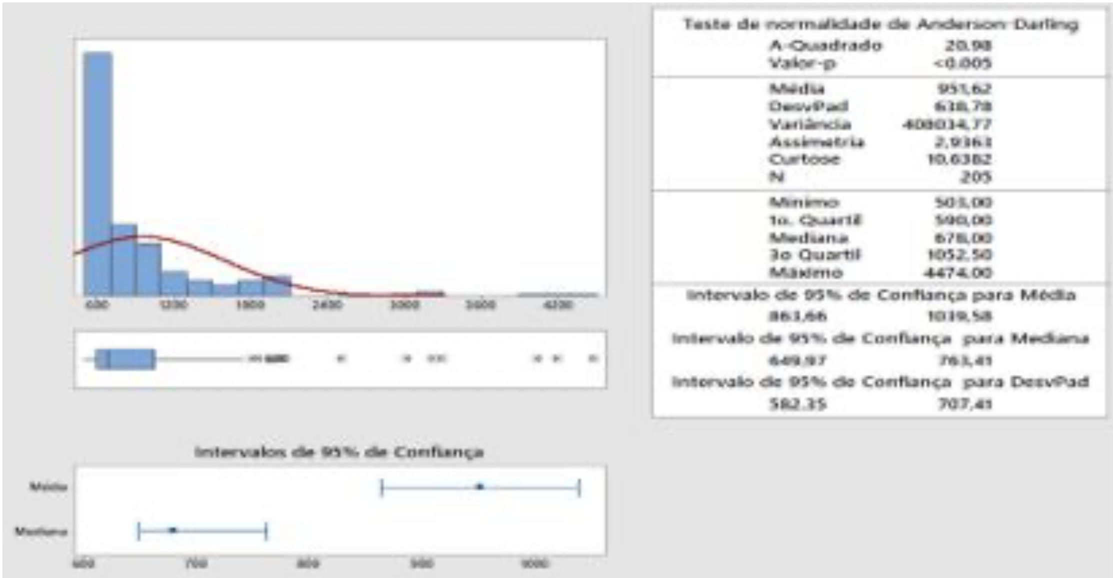


Figure 3
Descriptive analysis of the sample counts above 501 FP



After defining the scope of the analysis, the next step was ¨ Identification of the difference between the estimated and detailed counts (positive, negative or no difference) ¨. Table 6 presents the results of the execution of this step and it was found that most projects larger than 51FP, or approximately 87%, have an estimated count lower than the detailed count. The difference between estimated and detailed counts is identified by some authors. (Morrow, Wilkie & McChesney, 2014) and (Herlon & Dennis, FP Lite - An Alternative Approach to Sizing, 2011) suggest that aspects such as lack of in-depth knowledge of the system, change in scope, lack of documentation, some of the variables that impact this difference. In the case of our work, the difference points do not reflect these factors, since the estimated count was performed considering the detail. Thus, it can be inferred that the difference identified is specifically related to the detailed and estimated techniques.

Table 6
Quantitative counts with differences by FP range

Differences	51 to 200 FP		201 to 500 FP		above 501 FP	
	Number of counts	Percentage	Number of counts	Percentage	Number of counts	Percentage
No difference	56	2,52	8	1,62	2	0,97
Estimated > detailed	211	9,51	65	13,18	17	8,29
Estimated < detailed	1950	87,95	420	85,19	186	90,73
Totals	2217	100	493	100	205	100

In order to identify a pattern and to deepen the analysis, Table 7 presents the quantitative function points considering the differences between the estimated and the detailed identified. It is possible to identify that the percentage of difference between estimated and detailed is greater in the context of "smaller than detailed estimates." This fact reinforces the need to customize the estimated FPA, mainly in this scope, in order to become more adherent to the count detailed.

Table 7
Percentages of difference in FP between estimated and detailed
counts considering the scope of maintenance in systems larger than 51 FP

Ranges	Qty FP/Percentage	No difference	Estimated > Detailed	Estimated < Detailed
51 to 200 FP	Counting frequency	56	211	1950
	Detailed FP Qty	5113 FP	20129 FP	187866 FP
	Estimated FP Qty	5113 FP	22510 FP	146621 FP
	Percent Difference	0	11,82%	-21,95%
201 to 500 FP	Counting frequency	8	65	420
	Detailed FP Qty	2212 FP	19003 FP	129675
	Estimated FP Qty	2212 FP	20929 FP	101440
	Percent Difference	0	10,13%	-21,77%
above 501 FP	Counting frequency	2	17	186
	Detailed FP Qty	1027	16050 FP	178005 FP
	Estimated FP Qty	1027	16760 FP	140668 FP
	Percent Difference	0	4,42%	-20,97%

In order to analyze which functions have the most impact on the differences detected and to arrive at "Identifying patterns related to these differences", all the data functions and transactions involved were analyzed. Table 8 presents the analysis of the percentage of differences between estimated and detailed data, considering the amount of all types of data (ILF, EIF, EI, EO, EQ) involved in each maintenance project (above 51 FP).

It is interesting to note that a maintenance project can have more than one data type in its count. In this analysis, 15091 records corresponding to all counts greater than 51 FP were analyzed. In the range of 51 to 200 FP, we identified 29187 data functions and transactions involved in the detailed counts with differences (estimated and detailed), according to Table 8. Similar tables were developed and analysed for the range of 201 to 500 FP and the range above 501 FP.

Table 8
Percentages of frequency difference
of data type in the range 51 to 200 FP

Percent Difference	**Qty EIF**	**Qty ILF**	**Qty EQ**	**Qty EI**	**Qty EO**	**Total Qty Function**
-40,1 to -50,00	0	9	5	21	1	36
-30,1 to -40,00	0	144	367	4447	737	5695
-20,01 to -30,00	4	159	1517	4772	4997	11449
-10,1 to -20,00	0	134	1042	2310	1476	4962
-0,1 to -10	2	59	921	1550	642	3174
Total Estimated < detailed	6	505	3852	13100	7853	25316
0	0	6	87	106	41	240
Total estimated = detailed		6	87	106	41	240
0,1 to 10,00	0	26	486	669	271	1452
10,01 to 20,00	0	7	500	491	149	1147
20,01 to 30,00	0	1	272	285	187	745
30,1 to 40,00	0	0	212	75	0	287

Percent Difference	Qty EIF	Qty ILF	Qty EQ	Qty EI	Qty EO	Total Qty Function
Total estimated > detailed	0	34	1470	1520	607	3631
Total	6	545	5409	14726	8501	29187
Percent Total	0,02	1,87	18,53	50,45	29,13	

Considering this scope (Tables 8 and others), it is possible to identify that:

• From 83% to 90% of frequencies, approximately, refer to counts considering data types in which the estimate is less than detailed.

• Approximately less than 1% of frequencies have no difference between estimated and detailed

• From 9 to 16% of the frequencies, approximately, corresponds to counts in which the estimate is greater than the detailed one.

In addition, it is possible to verify that the data functions (ILF and EIF) are not the types of data that most impact the positive and negative differences found (approximately below 2% considering the total of the positive and negative frequencies). The transaction functions account for approximately 97% to 98% of the frequencies found. Specifically the EI function is the function that most generates differences (positive and negative) between estimated and detailed peaking approximately 50 to 56%.

It is still possible to identify that the EI represent (from 51 to 57% approximately) considering only the scope of the counts in which the estimate is smaller than the detailed one in relation to the total of negatives.

Considering the identified standard similar to the results of (Calazans, Martins, Masson, & Teixeira, 2017) and to attend the step "Evaluate the proposed metric customization (Calazans, Martins, Masson, & Teixeira, 2017) and, if feasible, the analysis of the available data "; was applied to the proposed customization (Calazans, Martins, Masson, & Teixeira, 2017), called Agile counting, in maintenance counts above 51 FP, as defined below:

• data functions by low complexity (ILF - 7FP; EIF - 5FP), similar to NESMA

• transaction functions:

- the EO and EQ by the medium complexity (EO - 5FP; EC - 4 FP), similar to NESMA
- the EI by the High Complexity (EI - 6FP), considering the results obtained by the research.

For proposal validation, the Agile count was applied in the 2,915 counts related to maintenance projects above 51 FP. It is interesting to note that (Calazans, Martins, Masson, & Teixeira, 2017) evaluated 10,405 counts for projects with less than 50 FP, with promising results. The initial objective was to verify the correlation between the estimated, detailed and the Agile counts (Table 9).

Table 9
Correlation between estimated, detailed and agile counts - 2915 counts

Ranges	Total FP detailed count	Total FP estimated count	Difference between estimated and detailed (%)	Total FP Agile Count	Difference between Agile and detailed (%)
51 to 200 FP	213108	174244	18,23	210512	1,21
201 to 500 FP	150890	124581	17,43	151123	-0,15
Above 500 FP	195082	158455	18,77	195765	-0,35

The result was very promising considering the large number of counts. The Agile counting approach presents a non-significant difference with the detailed count. In order to better analyze the result obtained, Table 10 show the maintenance counts above 51 FP per track, with the corresponding Estimated and Agile FP. Similar tables were developed and analysed for the range of 201 to 500 FP and the range above 501 FP.

Table 10
Agile vs. Estimated Proposal
Range 51 FP to 200 FP

Range FP	Qty of occurrences	Total FP detailed count	Total FP estimated count	Difference between estimated and detailed (%)	Total FP Agile Count	Difference between Agile and detailed (%)
51 to 60	425	23535	19052	-19,05	22884	-2,77
61 to 70	349	22830	18769	-17,79	22363	-2,05
71 to 80	274	20708	16598	-19,85	20324	-1,85
81 to 90	202	17263	14229	-17,58	17109	-0,89
91 to 100	179	17033	14107	-17,18	17071	0,22
101 to 110	136	14267	11675	-18,17	14185	-0,57
111 to 120	94	10836	9137	-15,68	11021	1,71
121 to 130	107	13392	11057	-17,44	13177	-1,61
131 to 140	86	11637	9534	-18,07	11438	-1,71
141 to 150	62	9054	7662	-15,37	9252	2,19
151 to 160	69	10731	8856	-17,47	10676	-0,51
161 to170	71	11717	9338	-20,30	11262	-3,88
171 to 180	63	11087	8840	-20,27	10980	-0,97
181 to 190	54	10003	8319	-16,83	10131	1,28
191 to 200	46	9015	7071	-21,56	8639	-4,17
Total	2217	213108	174244		210512	

Figure 4, 5 and 6 present the results, and make it possible to identify the closest proximity in almost all the ranges of the Agile model with the detailed count. Only in some bands this did not occur, for example the bands 201 to 220, 401 to 420, 701 to 750. But these bands do not have much representativity for their low quantitative occurrences, in addition those differences are below 8%.

Figure 4
Quantitative comparison of FP considering detailed, estimated counts and Agile model - range 51 to 200 FP.

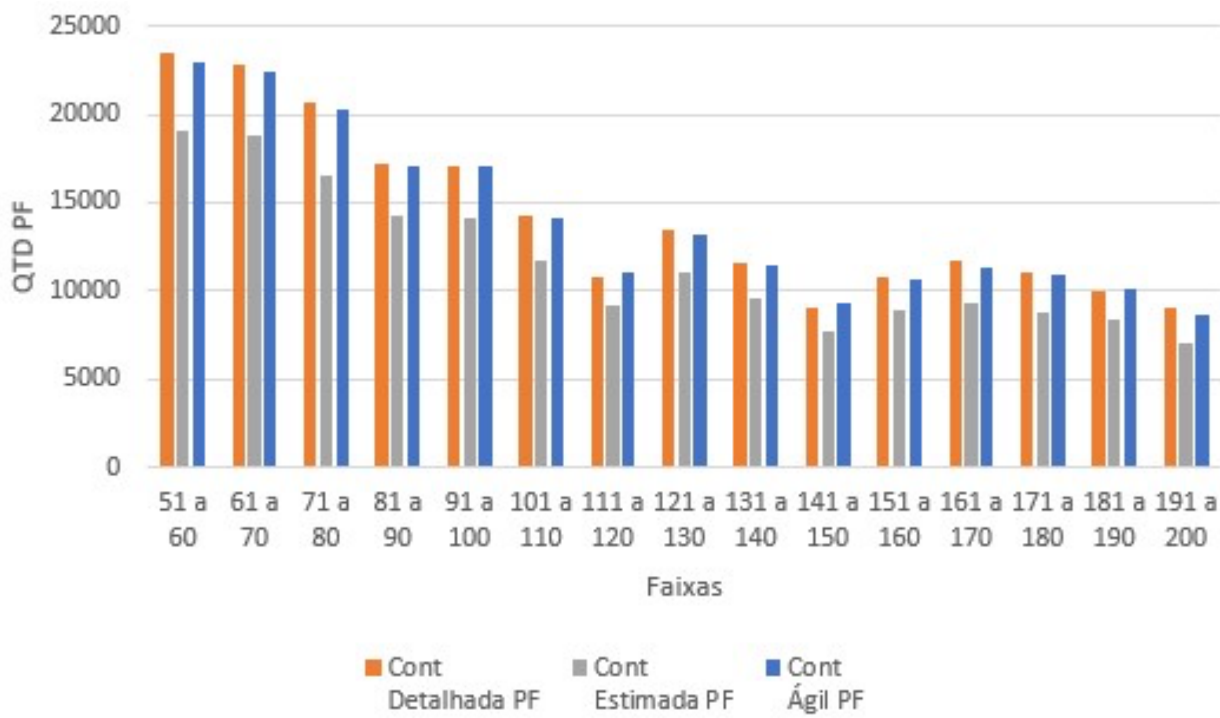


Figure 5
Quantitative comparison of FP considering detailed, estimated counts and Agile model - range 201 to 500 FP.

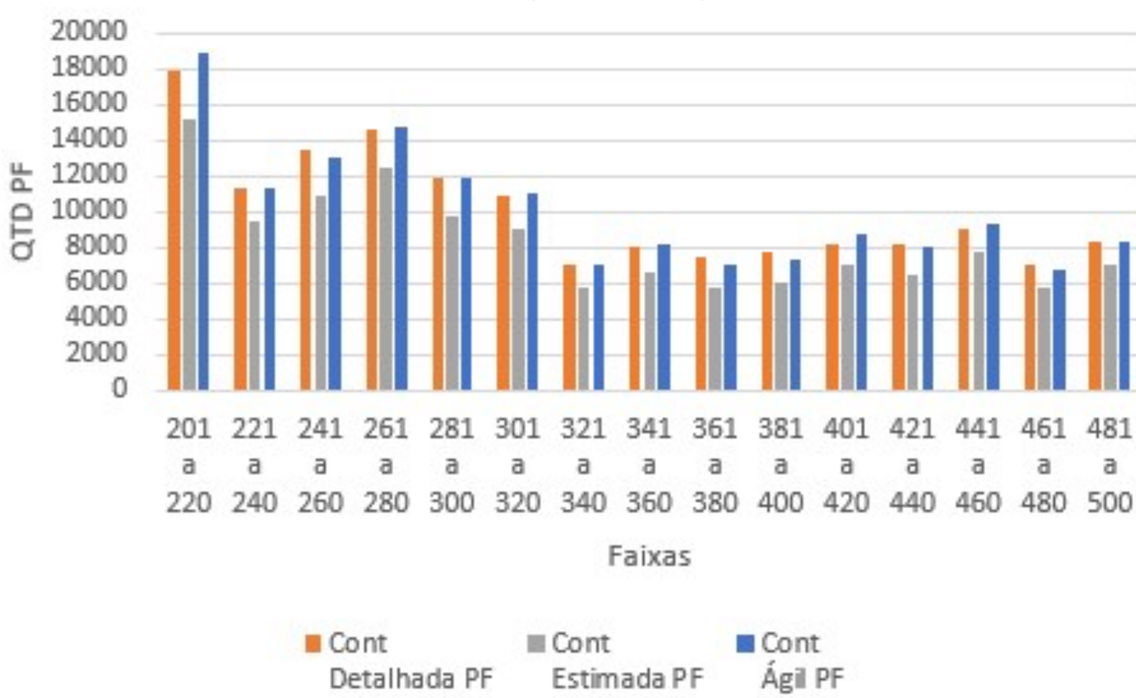
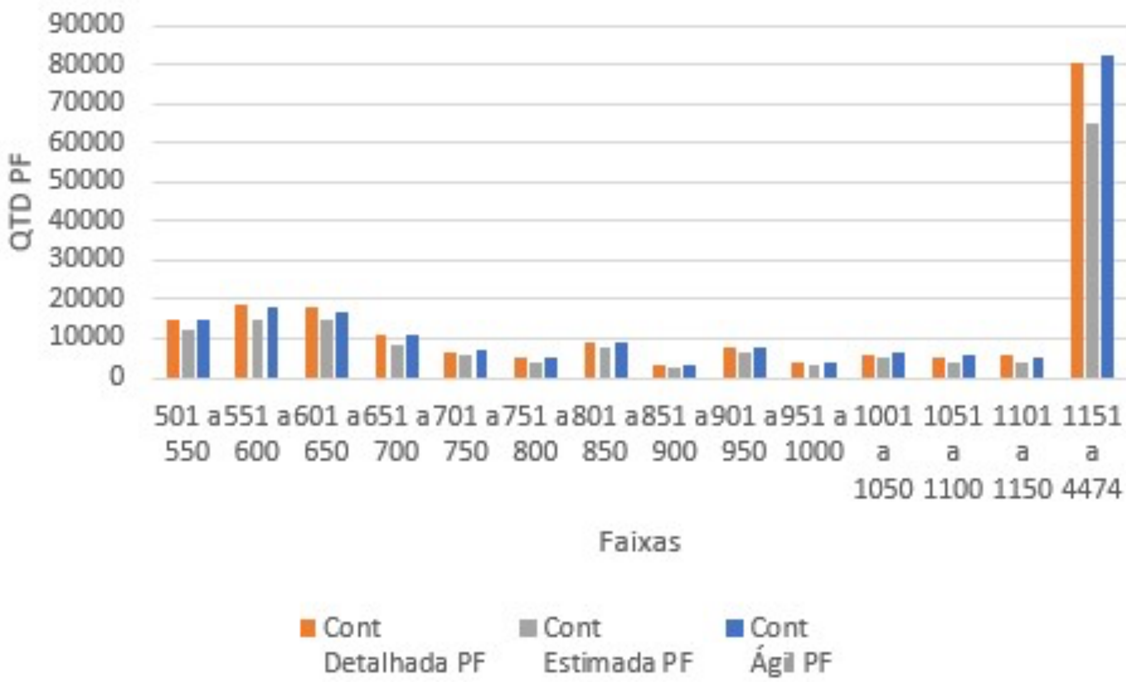


Figure 6
Quantitative comparison of FP considering detailed, estimated counts and model Agil - range 501 to 4474 FP.



In addition, it is important to note that, the Agile customization enables the elimination of the detailed counting of the counting process. That is, the effort of an estimated count and a detailed account, would be replaced by two estimated Agile scores. The Agile method would be applied at the beginning of the process for estimation and later (instead of the detailed count).

With the use of only the Agile Count, the effort, time and cost of the process of detailed counts would be reduced. Table 11 presents the projection of the estimated calculation of effort for counting considering the studied ranges and the bands below 50 FP according to the work of (Calazans, Martins, Masson, & Teixeira, 2017). In order to carry out these projections we considered the data of the authors (Herron, 2006) and (Morris, 2004).

Authors such as (Herron, 2006) suggest for detailed counts lower than 50 FP an approximate effort of 2:30 hours. That is, 10405 counts x 2:30, would be approximately 26,012 hours of detailed counting effort. These authors still suggest that in detailed counts of 50 to 150 FP an approximate effort of 4.3. Thus, for the range of 51 to 200 FP was considered 5 hours of effort per count. Similar projection was performed for counts of 201-500 FP, (Herron, 2006) suggest between 8.8 and 13.9. We opted to consider 10 hours. And for a range above 501 FP, 20 hours per detailed count was considered, since the counts analyzed in FP exceed the limit of 1000 FP.

(Morris, 2004) identifies that, for the non-detailed count, the productivity is 750 FP per day. Considering the day with 8 hours, we would have, approximately 94 FP per hour. Considering the Agile count, which is an adaptation of the estimate, and the effort quoted by (Morris, 2004), we would have the agile count below 50 FP performed in approximately 0.5 hour. This projection was considered for the other bands, for example from 51 to 200 FP were considered 2 hours for each estimated count; 201 to 500 FP, 5 hours and above 501, 10 hours.

We emphasize that the idea of the projection is to present a notion of the approximate amount of hours that would be reduced if Agile counting was used. It is clear that subsequently these data need to be ratified or rectified with actual measures.

Table 11
Estimation of effort of the counting process for the sample studied

Range	Qty Counts	Estimated Effort (hours)	Detailed Effort (hours)	Total Effort (detailed + Estimated) (hours)	Effort considering 2 Agile counts (hours)	Reduction Qty hours and percentage
Less than 50FP	10405	5.202	26.012	31.214	10.404	20.810 ….66,66%
51 to 200 FP	2217	4434	11085	15.519	8.868	6.651 42,85%
201 to 500 FP	493	2465	4930	7.395	4.930	2.465 33,33%
Above 501	205	2050	4100	6.150	4.100	2.050 33,33%
Total	13320	14151	46127	60.278	17.898	

Source: (Calazans, Martins, Masson, & Teixeira, 2017), adapted

Considering that the company in question outsources the counts, the costs of exchanging the counts for the Agile count would also be reduced. The analyzed company outsources the count by assigning differentiated values to estimated and detailed counts. Considering the projection of two Agile estimated counts instead of one Estimated and one Detailed, this would already provide a significant reduction of cost (Calazans, Martins, Masson, & Teixeira, 2017).

To validate the differences found, the statistical calculations or descriptive statistics are shown in Figure 7. These calculations aim to describe the sample and base the calculations required for later analysis of variance (ANOVA) and Tukey's test. The ANOVA allows to establish if the means of the populations under study are or are not, statistically equal, but it does not allow to detect which means are statistically different from the others. In order to verify which means differ, the Tukey test was used.

The Tukey test allows to establish the minimum significant difference, that is, the smallest difference of sample means that should be taken as statistically significant at a given level.

Figure 7
Detailed, Estimated and Agile descriptive statistics considering all counts above 51 FP

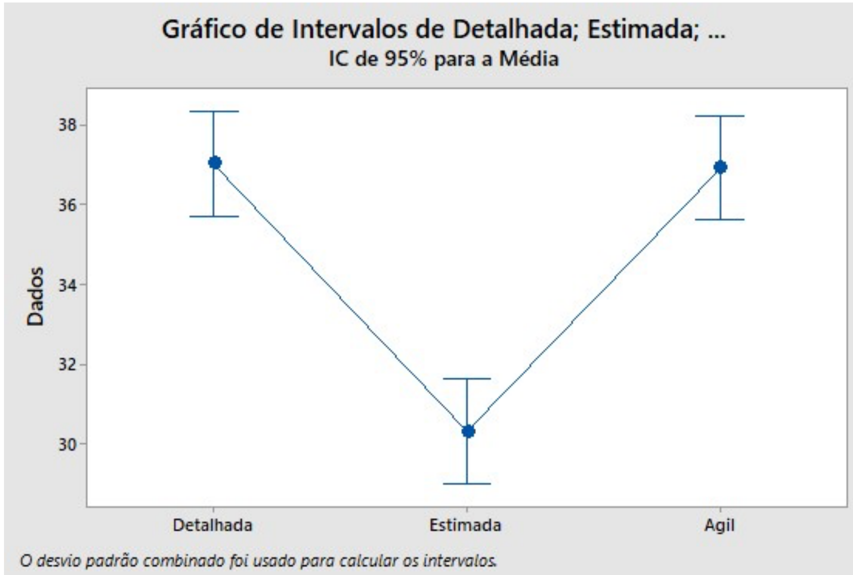
Estatísticas Descritivas: Detalhada; Estimada; Ágil

Estatísticas

Variável	N	N*	Média	EP Média	DesvPad	Mínimo	Q1	Mediana	Q3	Máximo
Detalhada	15091	0	37,05	0,738	90,63	3,00	7,00	15,00	36,00	3756,00
Estimada	15091	0	30,30	0,534	65,66	4,00	7,00	14,00	30,00	2504,00
Ágil	15091	0	36,94	0,720	88,39	4,00	7,00	15,00	36,00	3756,00

Figure 8 shows the detailed, Estimated, and Agile counting intervals graph. The interval plot displays the mean and confidence interval for each group. As can be observed, the detailed and Agile counts are well approximated.

Figure 8
Interval plot considering all counts above 51 FP



Two hypotheses were defined:

Null hypothesis All means are equal

Alternative hypothesis At least an average is different

The significance level was α = 0.05. Figure 9 presents the results, the p-value (0.000) indicates that there is sufficient evidence that not all the averages are the same when alpha is set to 0.05. In this analysis, the p-value calculation was used to reject the null hypothesis in favor of the alternative hypothesis. The p-value test is provided by computer statistical programs and in this test it is possible to have the value of the test t be in the theoretical distribution greater than the value obtained. Then, whenever the p-value is less than the established level of significance (in this study 0.05), the hypothesis that the means are equal is rejected.

In order to verify which means differ, the Tukey test was used. The Tukey test provides grouping information and 3 sets of multiple comparison confidence intervals. Figure 10 shows that group A contains the Detailed and Agile counts. While group B, the Estimated count. Only the Detailed and Agile counts share a letter and their methods are significantly approximate. This can also be seen in Figure 11, where the Agile-interval interval contains 0, demonstrating that the corresponding means do not differ from each other.

To complete the empirical analysis of the data, we chose to compare the Agile count with the proposals identified FPA Lite and FPA simple. The simple FPA can not be considered a custom FPA customization NESMA, since it creates new concepts. This approach, due to the new concepts addressed, can be considered a new metric.

Figure 9
Factor information -ANOVA

Informações dos Fatores

Fator	Níveis	Valores
Fator	3	Detalhada; Estimada; Agil

Análise de Variância

Fonte	GL	SQ (Aj.)	QM (Aj.)	Valor F	Valor-P
Fator	2	450381	225190	33,22	0,000
Erro	45270	306887137	6779		
Total	45272	307337517			

Sumário do Modelo

S	R2	R2(aj)	R2(pred)
82,3349	0,15%	0,14%	0,13%

Médias

Fator	N	Média	DesvPad	IC de 95%
Detalhada	15091	37,047	90,628	(35,734; 38,361)
Estimada	15091	30,302	65,657	(28,988; 31,615)
Agil	15091	36,936	88,390	(35,622; 38,250)

DesvPad Combinado = 82,3349

Figure 10
Tukey Comparisons

Comparações Emparelhadas de Tukey

Informações de Agrupamento Usando Método de Tukey e Confiança de 95%

Fator	N	Média	Agrupamento	
Detalhada	15091	37,047	A	
Agil	15091	36,936	A	
Estimada	15091	30,302		B

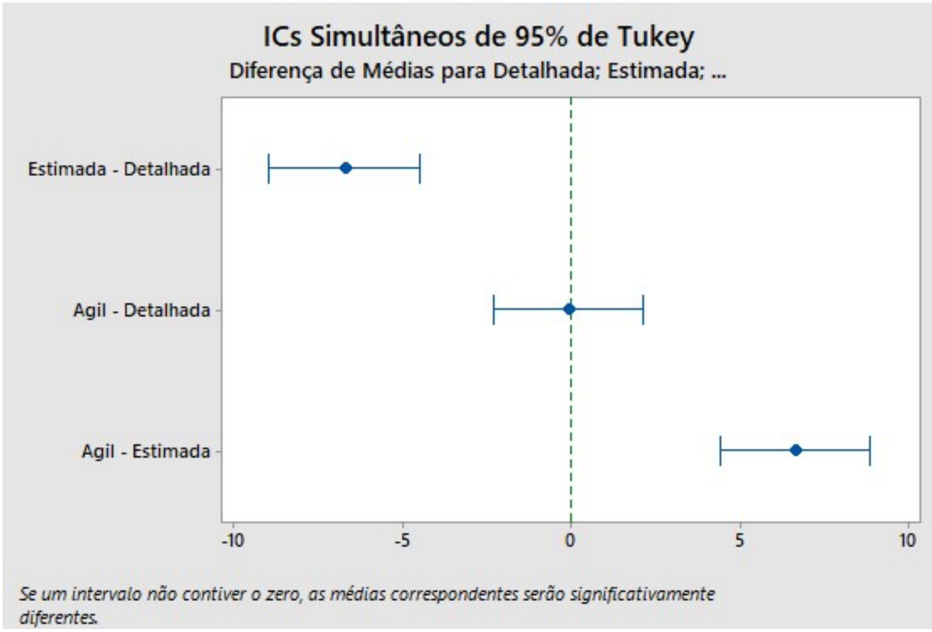
Médias que não compartilham uma letra são significativamente diferentes.

Testes Simultâneos de Tukey para as Diferenças de Médias

Diferença de Níveis	Diferença de Médias	EP da Diferença	IC de 95%	Valor-T	Valor-P Ajustado
Estimada - Detalhada	-6,746	0,948	(-8,964; -4,527)	-7,12	0,000
Agil - Detalhada	-0,111	0,948	(-2,330; 2,107)	-0,12	0,992
Agil - Estimada	6,634	0,948	(4,416; 8,853)	7,00	0,000

Nível de confiança individual = 98,07%

Figure 11
Simultaneous 95% Tukey ICs



FPA lite, as previously mentioned, customizes the concepts of the estimated Nesma, suggesting that the data functions are considered by the average complexity and not by the low completeness (ILF-10 FP and EIF-7 FP).Table 12 presents the approaches applied in this research.

Table 12
Approaches applied in research

Function Type	NESMA or Estimated	FPA lite	Agile Count	FPA Simple
ILF	7 FP	10 FP	7 FP	UGDG 7 FP
EIF	5 FP	7 FP	5 FP	UGDG 7 FP
EI	4 FP	4 FP	6 FP	UGEP 4,6 FP
SE	5 FP	5 FP	5 FP	UGEP 4,6
CE	4 FP	4 FP	4 FP	UGEP 4,6

Table 13 presents the results of the detailed, estimated, Agile and FPA Lite scores for counts above 51 FP.

Table 13
Results obtained with the agile count, FPA Lite, FPA Simple in counts above 51 FP

Range	Total FP detailed	Total FP Estimated	Difference between Detailed and Estimated (%)	Total FP Agile Count	Difference betwwen Agile Count and Detailed (%)	Total FPA Lite	Difference between FPA Lite and detailed (%)	FPA simple	Difference between FPA simple and detailed (%)
51 to 200 FP	213108	174244	18,23	210512	1,21	178824	16,09	185942,6	12,75
201 to 500 FP	150890	124581	17,43	151123	-0,15	126928	15,88	133679,4	11,41
Above 500 FP	195082	158455	18,77	195765	-0,35	160248	17,86	170014	12,85

As predicted and similar to the results from (Calazans, Martins, Masson, & Teixeira, 2017), the FPA Lite approach is not adequate for the studied sample of 2915 maintenance counts above 51 FP. We recall that most of the differences were identified in transactional functions rather than in data functions. The FPA Lite proposal increases the complexity of data functions.

On the other hand, the simple FPA decreases the size of the EO and improves the value of the EI. It raises the value of the IEA with respect to the detailed NESMA count. And for this sample of 2915 maintenance counts above 51 FP, the proposal is not adequate, similar to the results of (Calazans, Martins, Masson, & Teixeira, 2017) when it evaluated the counts below 50FP inclusive.

It is interesting to emphasize the growing need for studies with a representative amount of data to ratify or rectify existing proposals.

5. Conclusions

The objective of this work was to analyze the customization of a software size functional metric (called Agile count) focused on reducing the time, effort and cost of the maintenance project counting process and to test this approach in industry counts. For this, a bibliographical research was carried out to identify the existing proposals in the literature regarding the detailed FPA and estimated NESMA counts. Aspects related to the counting effort with the FPA were also identified.

Documentary research was conducted to identify the counting process used in some Brazilian public organizations and on the systems and projects that were measured.

Using the model (ISO / IEC 15939, 2007) to define the main research activities, we analyzed 2915 detailed maintenance counts of a large Brazilian public organization operating in the financial sector. This organization has a high level of IT outsourcing, even its metrics process is outsourced.

All counts were over 51 FP (inclusive). It was identified that from 83 to 90% of the counts, the estimated was smaller than the detailed count. And that transactional functions were the ones that most impacted the volume of differences between estimated and detailed.

Based on this finding, the Agile Count approach was applied, which adapts the estimated NESMA proposal and modifies only the complexity of a transaction function (EI). (Freitas Junior, Fantinato, & Sun, 2015) had already identified, in his study, the need to adjust weights and complexities. The approach was applied in the counts available above 51FP and the result was promising, similar to the studies of (Calazans, Martins, Masson, &

Teixeira, 2017). Variation lower than 1.21% in relation to the detailed count. Anova and the Tukey test were applied for statistical validation.

In order to reinforce the adequacy of the model, the other existing FPA Lite and Simple FPA approaches were applied to 2915 counts and the results were not comparable to the Agile approach proposed by the study.

To reduce the time, effort and cost of the maintenance project counting process, the Agile approach, by its adherence to the results of the detailed count, allows the substitution of the detailed count in the counting process. That is, the effort of the 1-count detail, would be replaced by the effort of the Agile score. The NESMA method or the Agile count would continue to be applied at the beginning of the development process for the estimation. With that the effort, time and cost of the counting process would be reduced. Effort projections were made considering authors such as (Herron, 2006) who demonstrated a substantial reduction in effort for counts of these projects. Considering that the company in question outsources the counts, the costs related to those hours would also be reduced.

Like other findings, we highlight some results of the research:

• It was identified that, of the total sample evaluated, counts with the objective of measuring improvement or maintenance correspond to 88.21%. Authors such as (Mohammad & Vinodani, 2014) suggest that maintenance consumes 60 to 80% of software.

• The research allowed to identify the greater instability of the transactional functions in relation to the data functions. That is, it has been identified that data functions impact counting changes less. This rectifies the proposal of (Herron & Dennis, 2011), since in its proposal it suggested the change, to a greater extent, in the complexity of the data functions.

• The substantial amount of small maintenance within a development process was another finding of the research. In the sample in question, of the 13,320 improvement or maintenance project counts, 10,405 had less than 50 FP. Authors such as (Jones, 2013) have already mentioned that the counting process is slow, and other authors, such as (Herron, 2006) suggest a 2.5-hour quantitative for detailed counting of minor projects that 50 FP.

It is up to Brazilian organizations to evaluate the need for a heavy or lighter counting process. The proposal of the Agile approach attempts to make this process more agile for small and large maintenance counts, reducing the effort and consequently the cost of these counts.

As future work it is interesting to evaluate the Agile proposal for the other ways of counting new development and application.

The FPA (detailed count) and the NESMA estimation proposal are considered to be well-founded models, but the work done shows that one can still analyze and identify improvement opportunities in these models, thus increasing the applicability and minimizing the disadvantages of these models.

Referencias

Badesul, B. D.-A. (2015). Edital. *Pregão Eletrônico N.º008/2015*. Rio grande do sul.

Brasil, A.-G. d. (2014). EDITAL. *PREGÃO ELETRÔNICO Nº 32/2014*. Brasilia.

Brasil, C. E. (2012). Contrato. *no. 4570/2012*. Brasilia.

Brasil, E. d. (2014). Edital. *Pregão Presencial Nº. 2011020.077/2014*. Minas Gerais.

Brasil, M. P.-P. (2015). Anexo I. *Termo de Referência*. Brasilia.

Calazans, A. M. (2017). Uma abordagem alternativa para contagem ágil do tamanho da manutenção do produto de software. *Revista Espacios, Vol. 38 (Nº 53)*, pp. 27-47.

Calazans, A. T., Paldes, R. A., & Mariano, A. (Jul/dez de 2015). Uma revisão sistemática da bibliografia sobre métricas funcionais de tamanho de software utilizando o enfoque meta analítico. (Uniceub, Ed.) *Universitas Gestão e TI, Brasília, v. 5, n. 2, p. 67-77, jul./dez. 2015, V. 5 N.2*, pp. 67-77. doi:10.5102/un.gti.v5i2.3532. https://www.revistaespacios.com/a17v38n53/a17v38n53p27.pdf

Ferrucci, F., Gravino, C., & Lavazza, L. (2016). Simple Function Points for Effort Estimation: a Further Assessment. *ACM*, pp. 1428-1433.

Freitas Junior, M. d., Fantinato, M., & Sun, V. (2015). Improvements to the Function Point Analysis Method: A Systematic Literature Review. *IEEE Transactions on engineering management, 4*, pp. 495-506.

Garmus, D. H. (2001). *Function point analysis: measurement practices for successful soft-ware projects. .* Boston: Addison-Wesley Longman Publishing Co.

Heeringen, H. v. (2015). *Software size measures and their use in software project estimation.* Acesso em 12 de 06 de 2017, disponível em Nesma: http://nesma.org/2015/05/software-size-measures-and-their-use-in-software-project-cost-estimation/

Herron, D. (2006). *Function Point Lite™ – Is It a Statistically VILFd Method of Counting?* David Consulting Group.

Herron, D., & Dennis, S. P. (2011). *FP Lite – An Alternative Approach to Sizing.* David Consulting Group.

Herron, D., & Dennis, S. P. (2011). *FP Lite™ – An Alternative Approach to Sizing.* David Consulting group.

ISO/IEC 15939. (2007). *Systems and software engineering -- Measurement process.* ISO/IEC.

ISO/IEC. (2002). ISO/IEC 20968:2002. *Software engineering - Mk II Function Point Analysis - Counting Practices Manual*.

ISO/IEC. (2005). ISO/IEC 24570:2005. *Software engineering -- NESMA functional size measurement method version 2.1 -- Definitions and counting guidelines for the application of Function Point Analysis*.

ISO/IEC. (2009). ISO/IEC 20926:2009. *Software and systems engineering -- Software measurement -- IFPUG functional size measurement method*.

ISO/IEC. (2011). ISO/IEC 19761:2011. *Software engineering -- COSMIC: a functional size measurement method*.

Jones, C. (2013). Function Points as a Universal Software Metric. *ACM SIGSOFT Software Engineering Notes, 38*.

Lavazza, L. (2015). Automated Function Points:Critical Evaluation and Discussion. (IEEE, Ed.) *6th International Workshop on Emerging Trends in Software Metrics*, pp. 35-43.

Marounek, P. (2012/2013). Simplified approach to effort estimation in software maintenance. *Journal of systems integration*, 51-63.

Matsutani, M. K., & Ribeiro, L. F. (2015). *Proposta de simplificação do processo de contagem de pontos de função na DATAPREV.* Datraprev -CGPP - Coordenação Geral de Programas e Projetos.

Meli, R. (2011). Simple Function Point: a new Functional Size Measurement Method fully compliant with IFPUG 4.x. *Software Measurement European Forum (SMEF)*.

Mohammad, I., & Vinodani, K. (2014). Development of a software maintenance cost estimation model: 4TH GL Perspective. *International Journal of Technical Research and Application, 6*.

Morris, P. (2004). *Níveis de contagens de pontos de função.* Total Metrics.

Morrow, P., Wilkie, F. G., & McChesney, I. R. (2014). Function point analysis using NESMA: simplifying the sizing without simplifying the size. (Springer, Ed.) *Software Qual J*, pp. 611–660.

Nguyen, V. (2010). Improved size and effort estimation models for software maintenance. *Dissertation*. CILFfornia: Faculty of the USC Graduate School University of Southern CILFfornia.

Raju, H. K., & Krishnegowda, Y. T. (may de 2013). Software Sizing and Productivity with Function Points. *Lecture Notes on Software Engineering, 1, n.2*. doi:10.7763/LNSE.2013.V1.46

SEI. (2010). CMMI for Development, Version 1.3. *CMMI for Development, Version 1.3: Software Engineering Institute, Carnegie Mellon University,"*.

SiFPA. (2014). Reference Manual SiFP-01.00-RMEN 01.01. *Simple Function Point Functional Size Measurement Method* . Acesso em 12 de 06 de 2017, disponível em http://www.sifpa.org/en/index.htm

Softex. (2016). MPS.BR - Melhoria de Processo do Software Brasileiro - Guia Geral MPS de Software.

Timp, A. (2015). *uTip - Early Function Point Analysis.* Acesso em 12 de 06 de 2017, disponível em Guidance from the Functional Sizing Standards Committee: http://www.ifpug.org/uTips/uTip003EarlyFPAandConsistentCostEstimating.pdf

Vazquez, C. E., Simoes, G. S., & Albert, R. M. (2013). *Análise de Pontos de Função:Medição, estimativas e gerenciamento de projetos de software* (13a. ed.). São Paulo: Érica.

Wang, X., Li, J., & Yu, F. (2008). Simplified function point analysis method aiming at small-to-mediumsized. *Jisuanji Gongcheng/Computer Engineering, 9*, pp. 103-105.