Systematic Review on the Use of Metrics for Estimating the Effort and Cost of Software Applicable to the Brazilian Public Sector

Washington Henrique Carvalho Almeida[1], Felipe Furtado[1], Luciano de Aguiar Monteiro[1], Fernando Escobar[2] and Sahra Karolina Gomes e Silva[3]
[1]Center for Studies and Advanced Systems of Recife – CESAR School
Recife, Brazil
[2]PMI-DF
Brasília, Brazil
[3]UNINASSAU
Teresina, Brazil
E-mail: {whca, fsfs, lam}@cesar.school
E-mail: {fernando.escobar.br, sahrask}@gmail.com

Abstract— **This article presents a systematic literature review concerning the use of metrics for estimating effort, cost, and timescale in the scope of software development services for the federal public administration sector, which seeks to obtain subsidies to reply to what metrics are used around the world and can be adopted within the Brazilian normative framework and applied to the sourcing of Information Technologies services. The systematic review is strongly related to the knowledge of associated literature, which can help us to understand the question. The research was conducted in some databases (AMC Digital Library, IEEE Xplore, Science Direct – Elsevier, Springer, Annals SBES and Annals SBQS) to which many filters were applied to obtain a set of articles that with thematic synthesis can highlight the adoption of expert-based estimation technique and metrics that address complexity. Finally, it was possible to find that there is truly little material related to the Brazilian case, which can highlight the importance of both systematic review and research.**

Keywords-systematic review; metrics; cost; effort.

I. INTRODUCTION

The evolution of Information Technology (IT) in the information era, boosted by the digital transformation of corporations, brings up several questions concerning the improvement of software quality. This is due to the amount of investment made by these corporations, whether they belong to the public or the private sector.

In Brazilian Federal Public Administration (FPA), software development is submitted to a very restrictive normative scope when it comes to setting delivery dates at the moment of hiring specialized services for these specific ends. With the advent of Normative Instruction IN 04/2008 and its constant alterations culminating in the current version, the IN 01/2019 from the Digital Government Department in the Ministry of Economy (DGD/ME) has equivalent legislation related to the other powers (Legislative and Judiciary). The IN 01/2019 regulates the requirements for hiring Information Technology services in the sphere of the Federal Executive Power. This period of time (2008-2019) and the adoption of metrics like the Function Point (FP) technique for compensating the hired effort have brought a several discrepancies that, in many cases, do not comprehend the real cost attributed to a commission.

Brazilian law requires payment for results but there is a discrepancy between the effort undertaken and the pricing process carried out by the contracting public institution [1].

Thus, as a way to fill this normative gap, this article aims to identify metrics for effort estimation used in software development projects with agile methods that seek to identify metrics or estimation processes that can be used to meet current Brazilian normative restrictions.

The remainder of this article is structured as follows: Section 2 approaches some concepts, which are essential for understanding the terminology that composes the scenario. Section 3 presents the review protocol, research conduction, and extraction results regarding this systematic review. In Section 4, we present the research results. Section 5 presents discussions, Section 6 the research limitations, and finally in Section 7, we present conclusions and future works.

II. CONCEPTS

Since the early days of Software Engineering, one of its fundamental problems is the estimation of effort, deadlines, and cost involved in software development. A lot has evolved in this area, but this key question is still the theme of some studies [2].

Earlier studies stated that large scale software development estimations and associated costs had a history of being more often wrong than right [3]. In this setting, several processes and metrics were established seeking to improve cost control [4], which is the basis of any area of Engineering.

Software measuring is concerned with the quantification of certain attributes in a software system, such as its complexity or its reliability. By comparing measured values among themselves and then to standards applied to an organization, it is possible to draw conclusions about the quality of the software or evaluate the efficiency of software processes, tools, and methods [5].

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

Software metrics aims to control and efficiently identify essential parameters that affect software development, as well as characteristics that cannot always be objectively measured. The term "software metrics" includes many activities that involve a certain level of software measurement and has a relationship with a series of concepts that base the adoption of metrics [2]. Some of these activities are listed as follows:

- models and measurements for estimating the cost and effort;
- data collection;
- models and quality measurements;
- models of reliability;
- security metrics;
- structural and complexity metrics;
- evaluation of the maturity of capacity;
- metrics management;
- evaluation of methods and tools;
- development by different teams of people.

In addition to that, the project and the analysis of software metrics are important in the life cycle of software development. Software metrics play a vital role in cost, quality, programming, reliability, and maintenance. There are many methods to decide what metrics must be used and for what ends [6]. The attributes of metrics can be either independent or they might depend on each other. In software engineering, there is not a consensus on what to measure and how to evaluate the result from these measurements [5].

Boehm [7], in his studies, assigned six (6) categories to the techniques to estimate the cost of a software system. This classification will be the basis for the thematic synthesis of metrics found in our selected studies.

The classification is defined as follows.

- **Based on the Opinion of Specialists**: this estimation is also known as an analogy-based estimate. It is the most used and it is generally accurate. The problem is that it is very subjective and can be biased. Techniques like Work Breakdown Structure (WBS) and techniques of group consensus like Delphi are used to eliminate the bias. Another deficiency is that the number of requirement alterations over time can render this method ineffective.
- **Based on Models**: there are many parametrical models but the most used one is COCOMO II. It is based on the assessment of various factors for estimates and it often needs dimension metrics such as Line of Code (LOC) – or its derivations, such as kLOC – or FP. The problem with these models is that they were designed having in mind a factory-like software development process based on a waterfall model.
- **Based on Regression**: linear regression is a statistical model where an equation estimates the expected value of a variable y given the values of some variables x. However, it has many deficiencies and needs a wide array of data. Another problem occurs in extreme cases, which are common in software engineering: usually, data used for building data clusters that will be tested in the equation are not collected properly due to limitations in time and budget.
- **Combined with Bayesian Statistics**: another alternative that attracts the methods of pure regression is a Bayesian approach, which combines the strengths of experience and methods based on regression. The Bayesian approach provides a formal process through which prior judgment by specialists can be combined with sampling (data) for producing a robust subsequent product. The Bayesian analysis is a method of inductive thinking that has been used in many scientific subjects.
- **Learning-Oriented**: a learning-oriented method is reasoning based on cases, in which it is possible to learn more adaptatively what cases in a sample of projects are better adjusted to the dominion application. It is currently based on machine learning and comes with Neural Networks methods, Genetic Algorithms, among others.
- **Based on Dynamic Systems**: techniques based on dynamics explicitly recognize that the effort applied to a software project or other factors of cost change throughout development; that is, they are dynamic rather than static. However, factors like deadlines, personnel level, project requirements, training needs, budget, etc. fluctuate over the course of development and it can cause fluctuations in the personal productivity of the project. This, in turn, has consequences on the probability of a project to be concluded within the planned deadline and budget – generally negative. System dynamics is a methodology of continuous modeling simulation in which the results and the behavior of the model are shown as information charts that change over time. The models are represented with modified networks with positive or negative feedback.

With the classification proposed, we will present COCOMO II, due to its wide adoption worldwide, and FP functional metrics, due to its wide application in service contraction for software development in Brazil [1].

A. COCOMO II

COCOMO II is a technique and tool for algorithmic modeling of costs. This empirical model was derived from the collection of data from various software projects of different sizes. These data were analyzed to discover formulas that would fit the observation in the best way. These formulas approached the system size and factors from the product, the project, the team, and the effort to develop a system [5].

COCOMO II was developed based on the first COCOMO cost estimation models (Constructive Cost Modelling), which were mostly based on the development of the original code [7]. This technique is usually linked to metrics and has four (4) basic models (application

composition model; early design model; reuse model; and, post-architecture model), depending on the metrics used, as seen in the FP and LOC studies.

B. Function Point Metrics

In 1979, Allan Albrecht, from International Business Machines (IBM), published a paper that brought to light a new metrics that, according to his experiences, proved itself effective for measuring software and posed as an alternative to metrics based on LOC. The above metrics started being used by many software companies as of the 1980s [8].

FP metrics were created from a principle stating that projects must be completed at a pre-established deadline, respecting the budget, and satisfying the client. From the beginning, it must have specific functional objectives and the desired value for money objectives. If the project can reach these objectives respecting the timetable and the budget, the client will be satisfied. Thus, it is necessary to measure productivity to identify and select the development systems and technologies that offer the most functionalities for application with the least effort and the lowest cost [1].

In Brazil, this technique has had accentuated growth, especially in the federal government sphere, with actions from Brazil's Federal Court of Accounts (TCU) and the publication of IN 02/2008 and IN 04/2008, both from the actual Ministry of Economy. It was determined that the services hiring should use the unit that would allow the measurement of results despite the existence of models other than the one standardized by International Function Point Users Group (IFPUG), and the fact that all of them are standardized by International Organization for Standardization (ISO). The IFPUG model is the most commonly used one in Brazil [1].

III. METHODS

A. Systematic Review of Planning

To understand the process adopted for conducting the systematic review, the following activities were defined, as shown in Figure 1:

- Formulate the research question: refers to define a question to support the research conduction;
- Define Research Protocol: regards to elaborate a protocol to research rules control.
- Search research bases: in this activity, a string is used to find studies in selected databases.
- Identify studies through title and abstract: refers to studies selection from reading titles and abstracts.
- Retrieve articles from databases: get the chosen studies from databases for more detailed analysis.
- Select studies according to the criteria: this activity includes selecting studies according to previously established criteria.
- Extract data: regards to gettting relevant information related to the research question.
- Evaluate quality: refers to quality assessment of the studies cited.

This way, in the first stage and step 1, the research question was formulated

1) What metrics adequately reward the effort applied in the construction of software functionality?

Complementarily, as secondary questions, which are inherently aligned with the answer to the main question, we have listed:

2) What metrics, according to the normative Brazilian framework, can be used to reward a supplier in cases where software development is outsourced by an FPA entity?
3) What metrics techniques are used in prompt methods and measure effort, deadline, cost, and size involved in software development?
4) Is FP Metrics used for calculating the payment of services in contracts outside of Brazil?

In the research protocol, we did define the Search Strings, databases to be consulted, and the criteria of quality for selecting the articles. Then, we move on to the Execution stage.

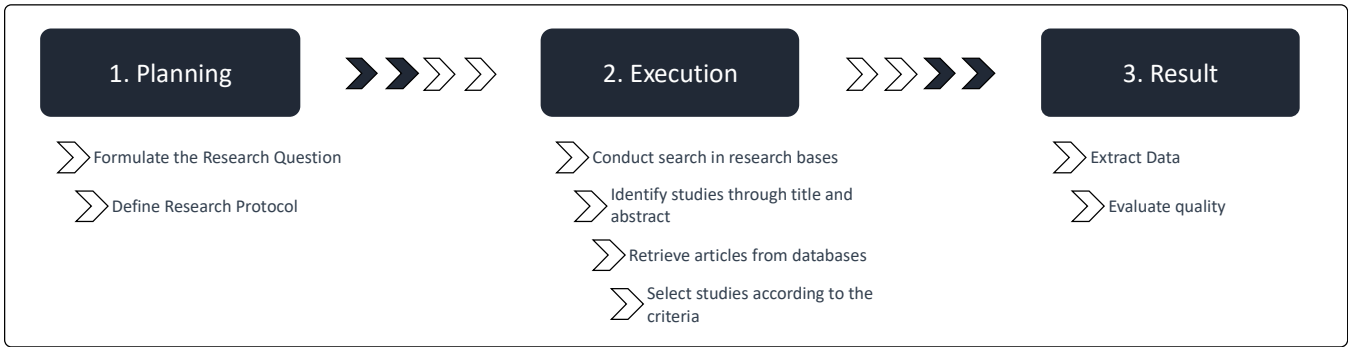

Figure 1. Diagram of the methodology of the systematic review.

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

B. Research Bases

The gathering of the articles was conducted on four bases with the automatic search strategy and two bases with the manual search strategy, as shown in Table I. Besides that, they were used due to the relevance of each base in this theme, by Kitchenham [4].

The literature repositories in the area seem to be promising. As one of the questions concerns a problem identified in Brazil, the study adopted the search from the Annals of Software Engineering Symposiums (SBES) and Software Quality (SBQS) held by the Brazilian Computer Society (SBC), which are reference events in the area and the theme of metrics is strongly based on Software Engineering and the studies of Software Quality.

TABLE I. SELECTED BASES.

Base	Address	Search
ACM Digital Library	https://dl.acm.org/	Automatic
IEEE Xplore	https://ieeexplore.ieee.org	Automatic
ScienceDirect – Elsevier	https://www.sciencedirect.com/	Automatic
Springer	https://link.springer.com/	Automatic
Annals SBES	The address is changed every year according to the organization of the event.	Manual
Annals SBQS	The address is changed every year according to the organization of the event.	Manual

In the manual searches in both events, there was a peculiarity. After a certain period, the books start being indexed to the ACM base, hence the manual research comprehended the years 2010 – 2019.

C. Research Strings

Considering such bases for research, some combinations of terms were fundamental for obtaining articles that would help systematic review and to obtain its state of the art.

We proceeded to cross the main keywords related to the themes we investigated, which were: “Smart Contract”, “Metric”, “Agile”, “Effort”, and “Cost”, in addition to other occasionally necessary ones for enriching our research sources, aiming to comprise a bigger amount of productions, avoiding the exclusion of a very important study or one that would stand out. Thus, some Search Strings were set up and all the selected papers referred to the 2010 – 2019 period. The Search Strings for each database are shown in Table II. In the initial stage, the number of articles found is shown in Table III.

TABLE II. SEARCH STRINGS.

Id	Database	Query applied
1	ACM Digital Library	[[All: "smart contract"] OR [All: metric]] AND [[All: "agile development"] OR [All: "agile"]] AND [[Abstract: "effort"] OR [Abstract: "cost"]] AND [[All: "smart contract"] OR [All: "metric"]] AND [Publication Date: (01/01/2010 TO 12/31/2019)]
2	IEEE Explore	((((“All Metadata":smart contract OR Metric) AND "All Metadata":"agile development" OR agile) AND "All Metadata":effort) AND "All Metadata":cost)
3	ScienceDirect – Elsevier	("smart contract" OR metric) AND ("agile development" OR agile) AND (effort OR cost) Abstract Effort OR cost
4	Springer	'("smart contract" OR metric) AND ("agile development" OR agile) AND (effort OR cost) AND "effort estimation"'

TABLE III. ARTICLES IN EACH BASE.

Database	Total
ACM Digital Library	185
IEEE Explore	61
Science Direct - Elsevier	813
Springer	161
Annals SBES	0
Annals SBQS	1
Total	**1221**

D. Criteria for Selection

Many criteria were selected so a certain article could be included to or excluded from the analysis for this research, such criteria are defined in Table IV and Table V. Inclusion criteria 5 was provided because the first FP contracts in Brazil were drawn in 2010. Exclusion criteria 3 refers to studies that are not entirely online accessible or fully inaccessible. The definition of exclusion criterion 7 is important to exclude studies that did not explain any metric for payment for services, such as FP, UCP, etc.

TABLE IV. CRITERIA FOR INCLUDING ARTICLES.

CI	Criteria for the inclusion of articles
1	Studies that show empirical or theoretical data or reports of experiences about metrics applied to payment based on the effort involved in the development of a software system;
2	Studies of quantitative and qualitative research;
3	Primary and secondary studies;
4	Studies wrote in English and Portuguese;
5	Studies published since 2010 [9].

TABLE V. CRITERIA FOR THE EXCLUSION OF ARTICLES.

CE	Criteria for the exclusion of articles
1	Repeated articles;
2	Similar articles;
3	Inaccessibility;
4	The article is not written in Portuguese or English;
5	Published as short paper or only as a poster;
6	Article without an abstract;
7	Studies did not focus on metrics for the payment of services;
8	Studies based solely on the opinion of specialists, not pointing to a specific experience;
9	Editorials, prefaces, forewords, article abstracts, interviews, news articles, analysis, tutorials, correspondence, discussions, commentaries, letters to readers, tutorial summaries, workshops, and panels.

E. Result Studies

After applying the criteria for inclusion and exclusion, we selected a set of studies that would be likely to answer the research question. At this stage, the articles were analyzed by

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

using the web application Rayyan, for cataloging the studies and sorting which ones were excluded and which were selected. The procedure described in the subsections above resulted in the number of articles per year and per database, as shown in Table VI.

In Table VII, we have a division by type, being 85 of the primary studies, 2 systematic mappings, 6 literature reviews, and 13 systematic reviews. Literature reviews are papers concerned with various metrics, but in their methodology, they do not show the thoroughness of a systematic review.

TABLE VI. ARTICLES PER BASE.

Database	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	TOTAL
ACM Digital Library	1	2	1	0	1	1	2	5	4	3	20
IEEE Explore	2	2	2	4	7	1	6	2	2	0	28
Science Direct - Elsevier	2	0	1	2	1	4	2	1	4	3	20
Springer	3	2	4	2	7	1	0	2	8	8	37
SBQS (Manual Research)	0	0	0	0	0	0	0	0	0	1	1
SBES (Manual Research)	0	0	0	0	0	0	0	0	0	0	0
Total											**106**

TABLE VII. TYPE OF STUDY.

Technique for Estimation	Quantity
Primary Study	85
Systematic Mapping	2
Literature Review	6
Systematic Review	13
Total	**106**

In terms of how these studies were published, we observed that 46 of these articles (43.4%) were presented in conferences, 35 (33.02%) were published in journals, 24 (22.64%) were chapters from books and 1 (0.94%) is a book, as seen in Figure 2.

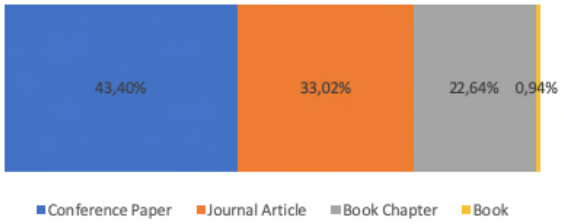

Figure 2. How the papers were published.

Among the 36 articles published in conferences, Figure 3 shows them, sorted according to their countries of origin.

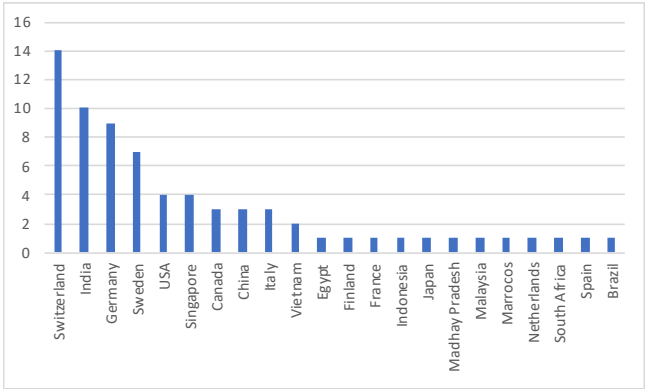

Figure 3. Countries where the conferences were held.

Moreover, the most relevant journals are listed ahead. 7 papers were published in the Journal of Systems and Software, 6 in Empirical Software Engineering, 5 in Information and Software Technology, 3 in Innovations in Systems and Software Engineering, 3 in Procedia Computer Science, 2 in the Journal of King Saud University – Computer and Information Sciences and Others with 1 article each, as shown in Table VIII.

TABLE VIII. ARTICLES SELECTED ACCORDING TO BASE.

Journal	Publisher	Quantity
Journal of Systems and Software	Elsevier	7
Empirical Software Engineering	Springer	6
Information and Software Technology	Elsevier	5
Innovations in Systems and Software Engineering	Springer	3
Procedia Computer Science	Elsevier	3
Journal of King Saud University - Computer and Information Sciences	Elsevier	2
Others (with 1 study)	ACM, Springer, and Elsevier	9
Total		**35**

After analyzing the articles and submitting them to a systematic synthesis with the classification of the estimation techniques, we found 6 articles that relate to the review of several techniques and 69 articles with primary studies that use various metrics, as shown in Table IX.

TABLE IX. CLASSIFICATION OF ARTICLES AND THE ESTIMATION TECHNIQUE THEY ADOPTED.

Estimation Technique	Quantity
Regression-Based	2
Model-Based	7
Learn-Based	20
Expert-Based	38
Dynamic-Based	2
Total	**69**

Amid this classification, we found the following metrics, though it is possible to observe that in some cases there is a combination of studies and several forms of metrics [10], and the combination of techniques like

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

COCOMO II and the metrics it naturally uses, as it is part of this model.

In addition to that, several studies related to metrics in agile methods, but they are complemented with some calibration done by using multiple factors or even machine learning, Bayesian statistics, neural networks, genetic algorithms, or other algorithms proposed in case studies. In the topic of discussions, we will present a review with the studies and techniques, methods, and approached metrics. The list of articles can be found in Appendix A, with the grade resulted from the classification as shown in Table X to the thematic synthesis also with techniques and metrics used.

TABLE X. TECHNIQUES AND METRICS FOUND.

Metrics	Kind of Measurement	Quantity
FP	Functional	10
COSMIC	Functional	4
UCP (Use Case Point)	Functional	5
SP (Stories Points)	Complexity	21
Velocity	Complexity	4
LOC or kLOC	Size	4

In this topic, we are bound to assess a narrative synthesis among 69 papers with classified metrics. This method builds a history based on the evidence found in the studies that were included [11].

According to Rodgers et al. [12], the recommended steps for conducting this synthesis are (i) development of theory; (ii) development of a preliminary synthesis; (iii) exploration of relationships inside and among studies; and (iv) assessment of the robustness of the product of synthesis. The robustness is presented in item 3.7 of the criteria and quality assessment.

F. Criteria of quality

Criteria of quality were adopted for classifying the results. The main goal when using quality criteria is to assess methodological aspects in the studies. When trying to assess the quality of the primary studies through quality criteria, the researcher seeks to increase reliability and generalization in the results [13].

Another way to measure quality in primary studies is through the application of a checklist, that is, a form that contains items that will be used to assess the quality of each study independently [11]. Therefore, a list was created for verifying the following criteria, as exposed in Table XI.

TABLE XI. QUALITY CRITERIA.

ID	Check List Item
1	Do the metrics adequately reward the effort applied to the construction of new software functionality?
2	Can the metrics be used following with the Brazilian normative framework for rewarding the supplier of software development is outsourced by an entity of the federal public administration?
3	Are the metrics used in agile methods that measure the effort, the deadline, the cost, and size involved in the construction of a software system?
4	Are the metrics used for rewarding services in contracts outside of Brazil?
5	Does the research show evidence or is it only a literature review?
6	Was there a detailed description of the review process?
7	Is the object of the research clearly defined?
8	Is there enough evidence to support a conclusion?
9	Does it show any charts, figures, or tables making a synthesis of the system?

Thus, a score of 0 or 1 was assigned in case the studies meet each of the 9 requirements, allowing the creation of a ranking. Out of the 106 studies initially selected, 69 were classified as likely to answer the questions – they are listed in Appendix A. Another 37 did not have a direct answer to the questions or showed inconclusive results.

G. Tools

To support the process, some tools had to be defined. Initially, the study used the Mendeley software for cataloging the list of articles yielded by the selected databases. For storing the articles (PDF) we used Zotero after the stages of selection of bases and list of articles have been repeated.

When the bases had been defined after the initial validation, the Rayyan software was used, which allowed the analysis of articles for the reading stage. The assessment of their quality was done employing an electronic form with questions and criteria for assessing each one of the selected articles.

IV. RESEARCH RESULTS

This research sought first to raise the metrics used in the industry for the adequate remuneration of the costs involved in the development of a software product, as presented, criteria were defined seeking to answer the research questions.

The research identified the most common metrics and various usage scenarios using the most varied systematics as a result, they were cataloged in the tables presented in the previous sections.

In Brazil, rework, which is common practice in agile methodologies, ends up not being properly remunerated because in the contractor's view it would be like paying for a job that does not deliver results. it is quite true that the rules and manuals of mandatory use due to the legislation seek to include rework when payment is by FP, but in practice, the problem lies in the imbalance that this type of practice ends up generating.

Also, the research helped to identify that the use of PF is not recommended for the support of systems, something that had already been identified in applied research in Brazil [14].

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

V. DISCUSSION

After the narrative and thematic syntheses, the following evidence was obtained to answer the research question:

Q1: Does the metrics adequately reward the effort applied to the construction of software functionality?

Yes, we found several metrics techniques that can assess the effort applied in the construction of software functionality, from parametric models like COCOMO [15]–[18] to its evolution COCOMO II [19][20] and this model requires a wide historical basis that is often based on functional measurement metrics like FP [21] and COSMIC [22]–[24] in addition to some studies that used LOC [25].

Q2: Can the metrics be used in following the Brazilian normative framework for rewarding the supplier of software development is outsourced by an entity of the federal public administration?

Yes, for functional measurement metrics, FP and COSMIC, but several metrics in more extensive studies using metrics applied to agile methods as Velocity [26], Sprint Points [27], Story Points [28], and Delivery Stories [29]. In some cases, it was combined with multiple factors techniques [30] to improve the precision and algorithms with verification list, even so, machine learning use [31].

Q3: Is the metric used in agile methods and measure effort, deadline, cost, and size involved in software development?

Yes, the same works presented in Q2 are about agile methods and measure these 3 aspects focused on software maintenance activities [32] and bugs fix [33].

Twenty studies focus on the use of machine learning techniques with the most diverse techniques since genetic algorithms [34], Bayesian statistics [35], fuzzy logic [36], [37], neural networks [38][39], and machine learning with multiple approaches [35][37][40][41].

Therefore, the most used are techniques based on expertise with an analogy (Expert-Based). It is presented in the studies several uses of the metrics in agile methods, mixing functional measure as FP already cited, or COSMIC [22]–[24][42] and classic agile metrics combined with multiple factors to precision calibration [26][27][43]–[45].

Q4: Is FP Metric used for calculating the payment of services in contracts outside of Brazil?

FP metric was found in 10 studies. The study of Russo et al. [46] is about FP used by the Italian public sector for critical service outsourcing. However, the metric is used to evaluate the functional size, deriving from this, productivity with effort and cost. Besides, the work explains Scrum Points that would be a fixed value of Hours inside a Sprint, for example, 40 hours, and the deliveries are made within an open scope system.

Another one [47], is about FP within a Dynamic-Model technique, a combination of Dynamic-Bases activity and Model-Based applied on agile development. But an old study from 2010 and another one uses COCOMO as a technique with Unadjusted Function Points (UFP) that would not be the FP use based on the IFPUG manual. The other studies [10][17][21][48]–[51] use analogy estimation mixing teams experience estimation with analogy and some agile metrics besides FP.

VI. RESEARCH LIMITATIONS

The Research looked for metrics used to pay for the effort and that can be adopted in Brazil following the Brazilian normative framework. The research found specific studies that dealt with the adoption of metrics in public organizations outside of Brazil.

Performing the automatic search in the databases, many of the studies did not meet the inclusion and exclusion criteria; therefore, after careful analysis, only 9% of the articles were selected. One of the bases returned 813 studies but most did not meet the criteria.

But one of the limitations is that the search for metrics in scientific works may not cover the practices developed by public organizations, so in an update of this systematic review, multi-vocal research should be adopted.

VII. CONCLUSION AND FUTURE WORKS

The most found metric in the studies was Story Points (which is based on a combination of the amount of effort involved in the development of a feature, with the complexity of that development, and the risk contained in it), very much in line with the development in agile methods, and which together with Velocity complements the metrics that address complexity.

Functional metrics, with a large advantage of Function Points (in the Brazilian case, in response to the regulations and guidelines of the control agencies), are second in the ranking. LOC, code size metric, performs last in studies, as it is a measure that we can consider linked to paradigms and technologies that are no longer in use.

Regarding the techniques, the predominance of Expert Based shows the importance of specialized opinion, with consideration and ponderation by Learn Based techniques, based on machine learning – very aligned with the data sciences. The grades attributed to the works, based on Quality Criteria, confirm this predominance.

The low number of works presented in Brazil contrasts with the importance of the theme for government hiring of these types of services, which constitutes an avenue of opportunities for future works.

Moreover, we can draw some conclusions concerning the research questions. The metrics related to complexity (Story points and Velocity) demonstrated to be more adequate to reward the effort applied in the construction of software functionality.

Also, we can infer some observations from Appendix A. Commonly, based on the defined technique (with a large predominance of Expert-based), the experiences adopt a combination of metrics. Together, they manage to better respond to the challenge of adequately reward the effort applied in the construction of software functionality.

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

ACKNOWLEDGMENT

The authors would like to thank CESAR School for financial support and especially teachers Alberto César Cavalcanti França and Ana Paula Cavalcanti Furtado who conducted the discipline of Systematic Review at the doctoral program.

REFERENCES

[1] W. H. C. Almeida and F. Furtado, "Análise sobre Métricas em Contratos de Fábricas de Software no âmbito da Administação Pública" (Analysis of metrics in software factory contracts within the public administration). 1ª ed, vol. 1. Rio de Janeiro, Albatroz, 2019.

[2] N. Fenton and J. Bieman, "Software Metrics: A Rigorous and Practical Approach", 3th Edition, CRC Press, 2014.

[3] R. E. Merwin, "Estimating software development schedules and costs", in Proceedings of the 9th Design Automation Workshop, New York, NY, USA, Jun. 1972, p. 1–4.

[4] B. Kitchenham, "What's up with software metrics? – A preliminary mapping study", Journal of Systems and Software, vol. 83, nº 1, p. 37–51, Jan. 2010.

[5] I. Sommerville, "Software engineering", 10th edition, Global edition, Pearson, 2016.

[6] N. Fenton and J. Bieman, "Software Metrics", vol. 1. Chapman and Hall/CRC, 2014.

[7] B. Boehm and K. Sullivan, "Software economics: Status and prospects", in Information & Software Technology, Nov 15, 1999, p. 937–946.

[8] R. Pressman and B. Maxim, "Engenharia de Software" (Software Engineering) - 8ª Edição. McGraw Hill Brasil.

[9] D. Kovags, F. L. Falchi, and A. R. Rivas, "Analysis of the Utilization of Scrum Framework Effort Estimation Metrics in Federal Public Administration", in Proceedings of the XVIII Brazilian Symposium on Software Quality - SBQS'19, Fortaleza, Brazil, 2019, p. 30–38.

[10] R. Popli and N. Chauhan, "Estimation in agile environment using resistance factors", in 2014 International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, Mar. 2014, p. 60–65.

[11] E. Y. Nakagawa, K. R. F. Scannavino, S. C. P. F. Fabbri, and F. C. Ferrari, "Revisão Sistemática da Literatura em Engenharia de Software: Teoria e Prática" (Systematic Review of Software Engineering Literature: Theory and Practice), 2017.

[12] M. Rodgers et al., "Testing Methodological Guidance on the Conduct of Narrative Synthesis in Systematic Reviews: Effectiveness of Interventions to Promote Smoke Alarm Ownership and Function", Evaluation, vol. 15, nº 1, p. 49–73.

[13] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature Reviews in Software Engineering", EBSE Technical report, Ver. 2.3., 2007.

[14] A. Trendowicz and R. Jeffery, "Software Project Effort Estimation", Software Project Effort Estimation, 2014.

[15] V. Nguyen, L. Huang, and B. Boehm, "An analysis of trends in productivity and cost drivers over years", in Proceedings of the 7th International Conference on Predictive Models in Software Engineering - Promise '11, Banff, Alberta, Canada, 2011, p. 1–10.

[16] S. Basri, N. Kama, F. Haneem, and S. A. Ismail, "Predicting effort for requirement changes during software development", in Proceedings of the 7th Symposium on Information and Communication Technology - SoICT '16, 2016, p. 380–387, Accessed: May 20, 2020. [Online].

[17] M. Farah-Stapleton, M. Auguston, and K. Giammarco, "Executable Behavioral Modeling of System and Software Architecture Specifications to Inform Resourcing Decisions", Procedia Computer Science, vol. 95, p. 48–57, 2016.

[18] J. A. Pow-Sang and R. Imbert, "Effort Estimation in Incremental Software Development Projects Using Function Points", vol. 340, T. Kim, C. Ramos, H. Kim, A. Kiumi, S. Mohammed, e D. Ślęzak, Orgs.

[19] R. Litoriya, N. Sharma, and A. Kothari, "Incorporating Cost driver substitution to improve the effort using Agile COCOMO II", in 2012 CSI 6th International Conference on Software Engineering (CONSEG), Indore, Madhay Pradesh, India, set. 2012, p. 1–7.

[20] S. Sunkle and V. Kulkarni, "Cost Estimation for Model-Driven Engineering", in Model Driven Engineering Languages and Systems, vol. 7590, R. B. France, J. Kazmeier, R. Breu, and C. Atkinson, Orgs. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, p. 659–675.

[21] J. Shah, N. Kama, and N. A. A. Bakar, "Estimating Change Effort Using a Combination of Change Impact Analysis Technique with Function Point Analysis", in Proceedings of the 2019 8th International Conference on Software and Information Engineering - ICSIE '19, Cairo, Egypt, 2019, p. 9–14, doi: 10.1145/3328833.3328836.

[22] I. Hussain, L. Kosseim, and O. Ormandjieva, "Towards Approximating COSMIC Functional Size from User Requirements in Agile Development Processes Using Text Mining", in Natural Language Processing and Information Systems, vol. 6177, C. J. Hopfe, Y. Rezgui, E. Métais, A. Preece, and H. Li, Orgs. 2010, p. 80–91.

[23] M. Salmanoglu, T. Hacaloglu, and O. Demirors, "Effort estimation for agile software development: comparative case studies using COSMIC functional size measurement and story points", in Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement on - IWSM Mensura '17, 2017, p. 41–49.

[24] A. Kaur and K. Kaur, "A COSMIC function points based test effort estimation model for mobile applications", Journal of King Saud University - Computer and Information Sciences, p. S131915781831317X, Mar. 2019.

[25] H. K. Sharma, R. Tomar, A. Dumka, and M. S. Aswal, "OpenECOCOMO: The algorithms and implementation of Extended Cost Constructive Model (E-COCOMO)", in 2015 1st International Conference on Next Generation Computing Technologies (NGCT), Dehradun, Sep. 2015, pp. 773-778.

[26] A. A. Mohallel and J. M. Bass, "Agile Software Development Practices in Egypt SMEs: A Grounded Theory Investigation", in Information and Communication Technologies for Development. Strengthening Southern-Driven Cooperation as a Catalyst for ICT4D, vol. 551, P. Nielsen and H. C. Kimaro, Orgs. Cham, 2019, p. 355–365.

[27] R. Popli and N. Chauhan, "A sprint-point based estimation technique in Scrum", in 2013 International Conference on

Information Systems and Computer Networks, Mathura, Mar. 2013, p. 98–103.

[28] J. Choudhari and U. Suman, "Story Points Based Effort Estimation Model for Software Maintenance", Procedia Technology, vol. 4, p. 761–765, 2012.

[29] M. Daneva et al., "Agile requirements prioritization in large-scale outsourced system projects: An empirical study", Journal of Systems and Software, vol. 86, nº 5, Art. nº 5, May 2013.

[30] M. Usman, R. Britto, L.-O. Damm, and J. Börstler, "Effort estimation in large-scale software development: An industrial case study", Information and Software Technology, vol. 99, p. 21–40, Jul. 2018.

[31] V. Khatibi Bardsiri, D. N. A. Jawawi, S. Z. M. Hashim, and E. Khatibi, "A flexible method to estimate the software development effort based on the classification of projects and localization of comparisons", Empir Software Eng, vol. 19, nº 4, p. 857–884, Aug. 2014.

[32] A. Espinoza and J. Garbajosa, "A study to support agile methods more effectively through traceability", Innovations in Systems and Software Engineering, vol. 7, nº 1, p. 53–69, Mar. 2011.

[33] S. Porru, A. Murgia, S. Demeyer, M. Marchesi, and R. Tonelli, "Estimating Story Points from Issue Reports", in Proceedings of the 12th International Conference on Predictive Models and Data Analytics in Software Engineering - PROMISE 2016, Ciudad Real, Spain, 2016, p. 1–10.

[34] S. Bilgaiyan, P. K. Panigrahi, and S. Mishra, "Chaos-Based Modified Morphological Genetic Algorithm for Effort Estimation in Agile Software Development", in A Journey Towards Bio-inspired Techniques in Software Engineering, vol. 185, J. Singh, S. Bilgaiyan, B. S. P. Mishra, and S. Dehuri, Orgs. 2020, p. 89–102.

[35] J. Khan, Z. A. Shaikh, and A. B. Nauman, "Development of Intelligent Effort Estimation Model Based on Fuzzy Logic Using Bayesian Networks", in Software Engineering, Business Continuity, and Education, vol. 257, T. Kim, H. Adeli, H. Kim, H. Kang, K. J. Kim, A. Kiumi, and B.-H. Kang, Orgs.

[36] A. Saini, L. Ahuja, and S. K. Khatri, "Effort Estimation of Agile Development using Fuzzy Logic", in 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, Aug. 2018.

[37] K. E. Rao and G. A. Rao, "Ensemble learning with recursive feature elimination integrated software effort estimation: a novel approach", Evol. Intel., Feb. 2020.

[38] Ch. Prasada Rao, P. Siva Kumar, S. Rama Sree, and J. Devi, "An Agile Effort Estimation Based on Story Points Using Machine Learning Techniques", in Proceedings of the 2nd International Conference on Computational Intelligence and Informatics, vol. 712, V. Bhateja, J. M. R. S. Tavares, B. P. Rani, V. K. Prasad, and K. S. Raju, Orgs. Singapore, 2018, p. 209–219.

[39] A. Panda, S. M. Satapathy, and S. K. Rath, "Empirical Validation of Neural Network Models for Agile Software Effort Estimation based on Story Points", Procedia Computer Science, vol. 57, 2015, p. 772–781.

[40] M. R. Tayyab, M. Usman, and W. Ahmad, "A Machine Learning Based Model for Software Cost Estimation", in Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016, vol. 16, Y. Bi, S. Kapoor, and R. Bhatia, Orgs.

[41] S. M. Satapathy and S. K. Rath, "Empirical assessment of machine learning models for agile software development effort estimation using story points", Innovations in Systems and Software Engineering, vol. 13, nº 2–3, p. 191–200, Sep. 2017.

[42] J.-F. Dumas-Monette and S. Trudel, "Requirements Engineering Quality Revealed through Functional Size Measurement: An Empirical Study in an Agile Context", in 2014 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement, Rotterdam, Netherlands, Oct. 2014, p. 222–232.

[43] S. Bilgaiyan, S. Mishra, and M. Das, "Effort estimation in agile software development using experimental validation of neural network models", International Journal of Information Technology, vol. 11, nº 3, p. 569–573, Sep. 2019.

[44] Mohd. Owais and R. Ramakishore, "Effort, duration and cost estimation in agile software development", in 2016 9th International Conference on Contemporary Computing (IC3), Noida, India, Aug. 2016, p. 1–5.

[45] W. Rosa, R. Madachy, B. Clark, and B. Boehm, "Early Phase Cost Models for Agile Software Processes in the US DoD", in 2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), Toronto, Canada, Nov 2017, pp. 30-37

[46] D. Russo, G. Taccogna, P. Ciancarini, A. Messina, and G. Succi, "Contracting agile developments for mission critical systems in the public sector", in the 40th International Conference, Gothenburg, Sweden, 2018.

[47] S. Kang, O. Choi, and J. Baik, "Model-Based Dynamic Cost Estimation and Tracking Method for Agile Software Development", in 2010 IEEE/ACIS 9th International Conference on Computer and Information Science, Aug. 2010, p. 743–748.

[48] H. Huijgens, A. van Deursen, L. L. Minku, and C. Lokan, "Effort and Cost in Software Engineering: A Comparison of Two Industrial Data Sets", in Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering - EASE'17, 2017, p. 51–60.

[49] E. G. Wanderley, A. Vasconcelos, and B. T. Avila, "Using Function Points in Agile Projects: A Comparative Analysis Between Existing Approaches", in Agile Methods, vol. 802, V. A. dos Santos, G. H. L. Pinto, and A. G. Serra Seca Neto, Orgs.

[50] W. Rosa, T. Packard, A. Krupanand, J. W. Bilbro, and M. M. Hodal, "COTS integration and estimation for ERP", Journal of Systems and Software, vol. 86, nº 2, Art. nº 2, Feb. 2013.

[51] M. Usman and R. Britto, "Effort Estimation in Co-located and Globally Distributed Agile Software Development: A Comparative Study", in 2016 Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), Berlin, Oct. 2016, p. 219–224.

[52] A. Kaushik, D. Kr. Tayal, and K. Yadav, "A Comparative Analysis on Effort Estimation for Agile and Non-agile Software Projects Using DBN-ALO", Arabian Journal for Science and Engineering, vol. 45, nº 4, Art. nº 4, Apr. 2020.

[53] L. L. Minku, “A novel online supervised hyperparameter tuning procedure applied to cross-company software effort estimation”, Empir Software Eng, vol. 24, nº 5, Art. nº 5, Oct. 2019.

[54] V. Khatibi Bardsiri, D. N. A. Jawawi, S. Z. M. Hashim, and E. Khatibi, “A PSO-based model to increase the accuracy of software development effort estimation”, Software Quality Journal, vol. 21, nº 3, Art. nº 3.

[55] S. Baker and E. Mendes, “Aggregating Expert-Driven Causal Maps for Web Effort Estimation”, in Advances in Software Engineering, vol. 117, T. Kim, H.-K. Kim, M. K. Khan, A. Kiumi, W. Fang, and D. Ślęzak, Orgs. Springer Berlin Heidelberg, 2010, p. 264–282.

[56] R. Popli and N. Chauhan, “Agile estimation using people and project related factors”, in 2014 International Conference on Computing for Sustainable Global Development (INDIACom), New Delhi, India, Mar. 2014, p. 564–569.

[57] S. Dragicevic, S. Celar, and M. Turic, “Bayesian network model for task effort estimation in agile software development”, Journal of Systems and Software, vol. 127, p. 109–119.

[58] R. Popli and N. Chauhan, “Cost and effort estimation in agile software development”, in 2014 International Conference on Reliability Optimization and Information Technology (ICROIT), Faridabad, Haryana, India, Feb. 2014, p. 57–61.

[59] M. Usman, K. Petersen, J. Börstler, and P. Santos Neto, “Developing and using checklists to improve software effort estimation: A multi-case study”, Journal of Systems and Software, vol. 146, p. 286–309, Dec. 2018.

[60] A. W. Md. M. Parvez, “Efficiency factor and risk factor-based user case point test effort estimation model compatible with agile software development”, in 2013 International Conference on Information Technology and Electrical Engineering (ICITEE), Yogyakarta, Indonesia, 2013.

[61] V. Lenarduzzi, I. Lunesu, M. Matta, and D. Taibi, “Functional Size Measures and Effort Estimation in Agile Development: A Replicated Study”, in Agile Processes in Software Engineering and Extreme Programming, vol. 212, C. Lassenius, T. Dingsøyr, and M. Paasivaara, Orgs.

[62] A. Zakrani, A. Najm, and A. Marzak, “Support Vector Regression Based on Grid-Search Method for Agile Software Effort Prediction”, in 2018 IEEE 5th International Congress on Information Science and Technology (CiSt), Marrakech, 2018.

[63] O. Malgonde and K. Chari, “An ensemble-based model for predicting agile software development effort”, Empirical Software Engineering, vol. 24, nº 2, Art. nº 2, Apr. 2019.

[64] D. Taibi, V. Lenarduzzi, M. O. Ahmad, and K. Liukkunen, “Comparing Communication Effort within the Scrum, Scrum with Kanban, XP, and Banana Development Processes”, in Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering - EASE’17, Karlskrona, Sweden, 2017, p. 258–263.

[65] A. Magazinius and R. Feldt, “Confirming Distortional Behaviors in Software Cost Estimation Practice”, in 2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), Oulu, Finland, Aug. 2011.

[66] K. Moharreri, A. V. Sapre, J. Ramanathan, and R. Ramnath, “Cost-Effective Supervised Learning Models for Software Effort Estimation in Agile Environments”, in 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC), Jun. 2016, p. 135–140.

[67] D. K. Goswami, S. Chakrabarti, and S. Bilgaiyan, “Effort Estimation of Web Based Applications Using ERD, Use Case Point Method and Machine Learning”, in Automated Software Engineering: A Deep Learning-Based Approach, vol. 8, 2020.

[68] E. Mendes, “Improving Software Effort Estimation Using an Expert-Centred Approach”, in Human-Centered Software Engineering, vol. 7623, M. Winckler, P. Forbrig, and R. Bernhaupt, Orgs.

[69] P. Ram, P. Rodriguez, and M. Oivo, “Software Process Measurement and Related Challenges in Agile Software Development: A Multiple Case Study”, in Product-Focused Software Process Improvement, vol. 11271, M. Kuhrmann, K. Schneider, D. Pfahl, S. Amasaki, M. Ciolkowski, R. Hebig, P. Tell, J. Klünder, and S. Küpper, Orgs. Cham, 2018, p. 272–287.

[70] E. Scott and D. Pfahl, “Using developers’ features to estimate story points”, in Proceedings of the 2018 International Conference on Software and System Process - ICSSP ’18, Gothenburg, Sweden, 2018, p. 106–110.

[71] S. R. Sree and C. P. Rao, “A Study on Application of Soft Computing Techniques for Software Effort Estimation”, in A Journey Towards Bio-inspired Techniques in Software Engineering, vol. 185, J. Singh, S. Bilgaiyan, B. S. P. Mishra, and S. Dehuri, Orgs.

[72] A. Effendi, R. Setiawan, and Z. E. Rasjid, “Adjustment Factor for Use Case Point Software Effort Estimation (Study Case: Student Desk Portal)”, Procedia Computer Science, vol. 157, p. 691–698, 2019.

[73] N. Ramasubbu and R. K. Balan, “Overcoming the challenges in cost estimation for distributed software projects”, in 2012 34th International Conference on Software Engineering (ICSE 2012), Zurich, 2012, pp. 91-101.

[74] T. Schulz, Ł. Radliński, T. Gorges, and W. Rosenstiel, “Predicting the Flow of Defect Correction Effort using a Bayesian Network Model”, Empirical Software Engineering, vol. 18, nº 3.

[75] P. Jodpimai, P. Sophatsathit, and C. Lursinsap, “Re-estimating software effort using prior phase efforts and data mining techniques”, Innovations in Systems and Software Engineering, vol. 14, nº 3, Art. nº 3, Sep. 2018.

[76] P. Pospieszny, “Software estimation: towards prescriptive analytics”, in The 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement, Gothenburg, Sweden, 2017, [Online]. Available: http://dl.acm.org/citation.cfm?doid=3143434.3143459.

[77] B. Tanveer, L. Guzmán, and U. M. Engel, “Understanding and improving effort estimation in Agile software development: an industrial case study”, in Proceedings of the International Workshop on Software and Systems Process - ICSSP ’16, Austin, Texas, 2016, p. 41–50.

[78] L. Kompella, “Advancement of Decision-Making in Agile Projects by Applying Logistic Regression on Estimates”, in 2013 IEEE 8th International Conference on Global

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-827-3

Software Engineering Workshops, Bari, Italy, Aug. 2013, p. 11–17.

[79] S. Thakur and H. Singh, "FDRD: Feature Driven Reuse Development Process Model", in 2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies, Ramanathapuram, 2014, pp. 1593-1598.

[80] F. Fellir, K. Nafil, R. Touahni, and L. Chung, "Improving Case Based Software Effort Estimation Using a Multi-criteria Decision Technique", in Software Engineering and Algorithms in Intelligent Systems, vol. 763, R. Silhavy, Springer International Publishing, 2019.

[81] S. Puhl and R. Fahney, "How to assign cost to avoidable requirements creep: A step towards the waterfall's agilization", in 2011 IEEE 19th International Requirements Engineering Conference, 2011, p. 307–312.

[82] M. Lusky, C. Powilat, and S. Böhm, "Software Cost Estimation for User-Centered Mobile App Development in Large Enterprises", in Advances in Human Factors, Software, and Systems Engineering, AHFE 2017. Advances in Intelligent Systems and Computing, vol 598. Springer, Chamvol, T. Ahram and W. Karwowski, Orgs. Cham, 2018.

APPENDIX A. ARTICLES, QUALITY CRITERIA, TECHNIQUES, AND METRICS

Ref.	Grade	Technique	Metric
[46]	7	Expert-Based	FP or SCRUM POINTS
[45]	7	Expert-Based	SP, UCP, FP, LOC, and OBJECT POINTS
[9]	7	Expert-Based	SP and UCP
[52]	6	Learn-Based	Data Set, DBN (Deep Belief Network) and ALO (Ant Lion Optimization)
[24]	6	Dynamic-Model	Data et, FP, SP and Algorithm
[30]	6	Expert-Based	Story Points to Maintenance
[40]	6	Learn-Based	ML (Machine Learning)
[47]	6	Expert-Based	COSMIC for Test
[28]	6	Learn-Based	Data Set and PSO
[53]	5	Model-Based	COCOMO and UFP (Unadjusted FP)
[54]	5	Learn-Based	SP and HH, ML
[27]	5	Learn-Based	Data Set, SP, and Multifactor
[55]	5	Expert-Based	EXPERT
[56]	5	Model-Based	Data Set and Statistical
[57]	5	Learn-Based	Data Set, Agile and COCOMO-II and ML and Data mining
[58]	5	Regression Based	Data Set, SP, and Velocity.
[20]	5	Model-Based	COCOMO II and Multifactor
[50]	5	Learn-Based	Data Set and ML
[59]	5	Dynamic-Model	Causal Structure Aggregation Model
[35]	5	Expert-Based	Multifactor and SP, Velocity
[60]	5	Expert-Based	Expert and Multifactor
[18]	5	Learn-Based	ML and Bayesian
[44]	5	Expert-Based	COSMIC
[41]	5	Expert-Based	Scrum adopted
[39]	5	Learn-Based	ML, NN, Fuzzy, Data Set
[37]	5	Expert-Based	FP, LOC, EOP (Enhanced Object Points for ERP)
[10]	5	Expert-Based	Sprint Points and Multifactor
[61]	5	Model-Based	PSO, Data Set, Algorithm
[19]	5	Expert-Based	UCP for Test
[62]	5	Learn-Based	Bayesian
[22]	5	Learn-Based	Data Set, and Algorithm
[29]	4	Expert-Based	FP
[26]	4	Expert-Based	EXPERT
[38]	4	Model-Based	SP, HH, EXPERT, Algorithm ensemble-based model
[63]	4	Learn-Based	SP, HH, EXPERT
[34]	4	Expert-Based	Data Set, Size (FP), Effort, Cost, and Duration
[64]	4	Expert-Based	COSMIC
[65]	4	Expert-Based	Data Set, User Stories, Story Points and Sprint Time
[66]	4	Model-Based	COCOMO and KLOC with Tool
[48]	4	Expert-Based	Interview and Multifactor
[23]	4	Expert-Based	COSMIC and SP, User Stories
[43]	4	Expert-Based	Velocity, Testing Performance, Issues' Estimation Accuracy, and Code Quality
[30]	4	Expert-Based	FP with Agile
[36]	4	Learn-Based	EXPERT
[67]	4	Expert-Based	EXPERT
[21]	4	Expert-Based	Velocity, SP
[68]	4	Learn-Based	ML and SP, Rede Neural.
[25]	4	Expert-Based	Scrum and FP, SP
[42]	4	Learn-Based	User Stories, Expertise, and Complexity using Fuzzy Logic to Predict
[69]	4	Expert-Based	Effort in Communication in Agile Environment
[70]	4	Expert-Based	UCP, size, and productivity
[49]	4	Learn-Based	Genetic Algorithm
[71]	3	Model-Based	COCOMO and KLOC
[72]	3	Expert-Based	UCP
[15]	3	Expert-Based	COCOMO and FP
[33]	3	Learn-Based	Data Set, Bayesian, and aspects for Maturity (CMMI)
[17]	3	Expert-Based	Maturity to Best Estimations
[73]	3	Learn-Based	NN, Fuzzy, and another ML
[74]	3	Expert-Based	Expert, Changes Requirements
[75]	3	Learn-Based	ML
[76]	3	Expert-Based	Data Set, Data Mining
[77]	3	Expert-Based	SP for Issues
[78]	2	Expert-Based	Multifactor (RF, RNF, and DP-Domain Properties)
[79]	2	Learn-Based	COCOMO and Change Request
[80]	2	Regression Based	Logistic Regression Model
[16]	2	Expert-Based	FDD and Reuse
[81]	1	Expert-Based	Delphi
[82]	1	Expert-Based	Data Set and Quality of Requirements