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].
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 getting 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.](image-url)
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>
<thead>
<tr>
<th>Table I. Selected Bases.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Base</strong></td>
</tr>
<tr>
<td>ACM Digital Library</td>
</tr>
<tr>
<td>IEEE Xplore</td>
</tr>
<tr>
<td>ScienceDirect – Elsevier</td>
</tr>
<tr>
<td>Springer</td>
</tr>
<tr>
<td>Annals SBES</td>
</tr>
<tr>
<td>Annals SBQS</td>
</tr>
</tbody>
</table>

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>
<thead>
<tr>
<th>Table II. Search Strings.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Id</strong></td>
</tr>
</tbody>
</table>

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>
<thead>
<tr>
<th>Table III. Articles in Each Base.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Database</strong></td>
</tr>
<tr>
<td>ACM Digital Library</td>
</tr>
<tr>
<td>IEEE Explore</td>
</tr>
<tr>
<td>Science Direct - Elsevier</td>
</tr>
<tr>
<td>Springer</td>
</tr>
<tr>
<td>Annals SBES</td>
</tr>
<tr>
<td>Annals SBQS</td>
</tr>
<tr>
<td><strong>Total</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table IV. Criteria for Including Articles.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CI</strong></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table V. Criteria for the Exclusion of Articles.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CE</strong></td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
</tbody>
</table>

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...
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.

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.

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.

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

![Conference Paper](image1.png)

![Journal Article](image2.png)

![Book Chapter](image3.png)

![Book](image4.png)

**Figure 2.** How the papers were published.

![Countries where the conferences were held](image5.png)

**Figure 3.** Countries where the conferences were held.

In Table IX, we found the following metrics: 2 Regression-Based, 7 Model-Based, 20 Learn-Based, 38 Expert-Based, 2 Dynamic-Based, and 69 Total.

![Classification of articles and the estimation technique they adopted](image6.png)

**Figure 4.** Classification of articles and the estimation technique they adopted.
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.**

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Kind of Measurement</th>
<th>Quantity</th>
</tr>
</thead>
<tbody>
<tr>
<td>FP</td>
<td>Functional</td>
<td>10</td>
</tr>
<tr>
<td>COSMIC</td>
<td>Functional</td>
<td>4</td>
</tr>
<tr>
<td>UCP (Use Case Point)</td>
<td>Functional</td>
<td>5</td>
</tr>
<tr>
<td>SP (Stories Points)</td>
<td>Complexity</td>
<td>21</td>
</tr>
<tr>
<td>Velocity</td>
<td>Complexity</td>
<td>4</td>
</tr>
<tr>
<td>LOC or kLOC</td>
<td>Size</td>
<td>4</td>
</tr>
</tbody>
</table>

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.**

<table>
<thead>
<tr>
<th>ID</th>
<th>Check List Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Do the metrics adequately reward the effort applied to the construction of new software functionality?</td>
</tr>
<tr>
<td>2</td>
<td>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?</td>
</tr>
<tr>
<td>3</td>
<td>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?</td>
</tr>
</tbody>
</table>

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].
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.
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


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

<table>
<thead>
<tr>
<th>Ref.</th>
<th>Grade</th>
<th>Technique</th>
<th>Metric</th>
</tr>
</thead>
<tbody>
<tr>
<td>[46]</td>
<td>7</td>
<td>Expert-Based</td>
<td>FP or SCRUM POINTS</td>
</tr>
<tr>
<td>[45]</td>
<td>7</td>
<td>Expert-Based</td>
<td>SP, UCP, FP, LOC, and OBJECT POINTS</td>
</tr>
<tr>
<td>[9]</td>
<td>7</td>
<td>Expert-Based</td>
<td>SP and UCP</td>
</tr>
<tr>
<td>[52]</td>
<td>6</td>
<td>Learn-Based</td>
<td>Data Set, DBN (Deep Belief Network) and ALO (Ant Lion Optimization)</td>
</tr>
<tr>
<td>[24]</td>
<td>6</td>
<td>Dynamic-Model</td>
<td>Data et, FP, SP and Algorithm</td>
</tr>
<tr>
<td>[30]</td>
<td>6</td>
<td>Expert-Based</td>
<td>Story Points to Maintenance</td>
</tr>
<tr>
<td>[40]</td>
<td>6</td>
<td>Learn-Based</td>
<td>ML (Machine Learning)</td>
</tr>
<tr>
<td>[47]</td>
<td>6</td>
<td>Expert-Based</td>
<td>COSMIC for Test</td>
</tr>
<tr>
<td>[22]</td>
<td>6</td>
<td>Learn-Based</td>
<td>Data Set and PSO</td>
</tr>
<tr>
<td>[51]</td>
<td>5</td>
<td>Model-Based</td>
<td>COCOMO and UFP (Unadjusted FP)</td>
</tr>
<tr>
<td>[54]</td>
<td>5</td>
<td>Learn-Based</td>
<td>SP and HH, ML</td>
</tr>
<tr>
<td>[27]</td>
<td>5</td>
<td>Learn-Based</td>
<td>Data Set, SP, and Multifactor</td>
</tr>
<tr>
<td>[55]</td>
<td>5</td>
<td>Expert-Based</td>
<td>EXPERT</td>
</tr>
<tr>
<td>[56]</td>
<td>5</td>
<td>Model-Based</td>
<td>Data Set and Statistical</td>
</tr>
<tr>
<td>[57]</td>
<td>5</td>
<td>Learn-Based</td>
<td>Data Set, Agile and COCOMO-II and ML and Data mining</td>
</tr>
<tr>
<td>[58]</td>
<td>5</td>
<td>Regression Based</td>
<td>Data Set, SP, and Velocity.</td>
</tr>
<tr>
<td>[20]</td>
<td>5</td>
<td>Model-Based</td>
<td>COCOMO II and Multifactor</td>
</tr>
<tr>
<td>[50]</td>
<td>5</td>
<td>Learn-Based</td>
<td>Data Set and ML</td>
</tr>
<tr>
<td>[59]</td>
<td>5</td>
<td>Dynamic-Model</td>
<td>Causal Structure Aggregation Model</td>
</tr>
<tr>
<td>[35]</td>
<td>5</td>
<td>Expert-Based</td>
<td>Multifactor and SP, Velocity</td>
</tr>
<tr>
<td>[60]</td>
<td>5</td>
<td>Expert-Based</td>
<td>Expert and Multifactor</td>
</tr>
<tr>
<td>[18]</td>
<td>5</td>
<td>Learn-Based</td>
<td>ML and Bayesian</td>
</tr>
<tr>
<td>[44]</td>
<td>5</td>
<td>Expert-Based</td>
<td>COSMIC</td>
</tr>
<tr>
<td>[41]</td>
<td>5</td>
<td>Expert-Based</td>
<td>Scrum adopted</td>
</tr>
<tr>
<td>[39]</td>
<td>5</td>
<td>Learn-Based</td>
<td>ML, NN, Fuzzy, Data Set</td>
</tr>
<tr>
<td>[37]</td>
<td>5</td>
<td>Expert-Based</td>
<td>FP, LOC, EOP (Enhanced Object Points for ERP)</td>
</tr>
<tr>
<td>[10]</td>
<td>5</td>
<td>Expert-Based</td>
<td>Sprint Points and Multifactor</td>
</tr>
<tr>
<td>[61]</td>
<td>5</td>
<td>Model-Based</td>
<td>PSO, Data Set, Algorithm</td>
</tr>
<tr>
<td>[19]</td>
<td>5</td>
<td>Expert-Based</td>
<td>UCP for Test</td>
</tr>
<tr>
<td>[62]</td>
<td>5</td>
<td>Learn-Based</td>
<td>Bayesian</td>
</tr>
<tr>
<td>[22]</td>
<td>5</td>
<td>Learn-Based</td>
<td>Data Set, and Algorithm</td>
</tr>
</tbody>
</table>