IET Research Journals

Software development effort estimation: A systematic mapping study

The Institution of Engineering and Technology

Software development effort estimation: A systematic mapping study

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Carlos Eduardo Carbonera[1], *Kleinner Farias*[1*], *Vinicius Bischoff*[1]

[1] *Applied computing graduate program (PPGCA), University of Vale do Rio dos Sinos, São Leopoldo, RS, Brazil*
** E-mail: kleinnerfarias@unisinos.br*

Abstract: *Context*: The field of software-development effort estimation explores ways of defining effort through prediction approaches. Even though this field has a crucial impact on budgeting and project planning in industry, the number of works classifying and examining currently available approaches is still small. *Objective*: This article, therefore, presents a comprehensive overview of these approaches, and pinpoints research gaps, challenges, and trends. *Method*: A systematic mapping of the literature was designed and performed based on well-established practical guidelines. In total, 120 primary studies were selected, analyzed and categorized, after applying a careful filtering process from a sample of 3,746 candidate studies to answer six research questions. *Results*: Over 70% of the selected studies adopted multiple effort estimation approaches; over 45% adopted evaluation research as research method; over 90% of the participants were students, rather than professionals; most studies had their quality assessed as high, and were most commonly published in journals. *Conclusions*: Our study benefits practitioners and researchers by providing a body of knowledge about the current literature, serving as a starting point for upcoming studies. This article reports challenges worth investigating, regarding the use of cognitive load and team interaction.

1 Introduction

The estimation of effort development can be briefly defined as a set of tasks that should be performed to create estimates, which are usually expressed in terms of hours or money [1–4]. Many approaches of effort estimation were proposed in the past few decades, mainly to support managers and developers while performing software development tasks. Planning Poker [5], COCOMO [6], Delphi [7], and multi-objective software effort estimation [18] would be examples of these approaches, which are often used by project managers to elaborate budget, forecast iteration plans, and define project plans.

Even though estimating effort has a crucial impact on budgeting and project planning in industry, it is still an open question. This means that practitioners and researchers do not have an effective and widely adopted approach. Consequently, they still need to select one approach, among currently available approaches, that best fits their needs. If an estimation approach does not fit their needs, then the adoption of this approach becomes questionable in realistic scenarios.

In this sense, researchers and practitioners need to survey the current literature, being an inherently manual, error-prone task. Moreover, works classifying and examining the literature are still scarce. They end up not having a mapping of the literature considering recent works. As a result, some important questions remain without answers: Which cost-drivers have been most commonly used? What are the most commonly used research methods? Which research issues have been investigated more frequently? Who participates in the studies? How can the current studies be qualitatively assessed? Where have the current studies been published?

Recent literature reviews aim to gather findings about: (1) a new view on best practices in model-based effort estimation [8]; (2) a basis for the improvement of software estimation research [9]; (3) a summary of estimation knowledge through a review of surveys on software effort estimation [P30]; and (4) an extensive review of studies on the expert-judgment-based estimation of software development effort [P31]. Although the results introduced by these studies reveal the maturity of particular topics related to software development effort estimation, they do not provide a systematic map of the literature, and draw some research directions and trends. Instead, they focus on specific facets about effort estimation, such as cost estimation based on expert judgment.

This article, therefore, presents a comprehensive overview and understanding of the literature (Section 4), as well as pinpoints research gaps, challenges, and trends (Section 5). A systematic mapping of the literature was designed and performed based on well-established practical guidelines [9–12]. In total, 120 primary studies were selected, analyzed and categorized after applying a careful filtering process from a sample of 3,767 studies to answer six research questions. We chose systematic mapping study as research method to address our research questions because of some reasons. First, it provides a wide overview of a research area and establishes if empirical evidence exists on a topic, and if this evidence is (or can be) quantified. Second, it summarizes the current literature and highlights some challenges and directions that can be explored in upcoming studies. Third, it discusses which research topics still require an in-depth analysis and synthesis [13–17].

The main contributions of our article are: (1) a body of knowledge through a systematic map about the current literature, which serves as a starting point for future works of Ph.D. students, and encourages researchers to explore some promising challenges; (2) a review protocol that can be reused in future research; and (3) a comprehensive overview and quality assessment of the current studies. In addition, this article reduces the learning curve and bias to perform upcoming literature reviews, and uncovers when and where the most representative studies have been published. It also shows how such studies are distributed over the last years.

The remainder of the paper is organized as follows: Section 2 introduces our review protocol. Section 3 explains the adopted procedures to filter potentially relevant studies, and Section 4 presents the obtained results. Section 5 introduces discussions and draws some future directions. Section 6 introduces the adopted procedures to minimize threats to validity. Section 7 compares this study with the current literature. Finally, Section 8 presents some concluding remarks and future work.

2 Planning

This section aims to outline our review protocol. Figure 1 introduces the adopted systematic mapping process, which is formed by three phases composed by a set of activities and artifacts, including

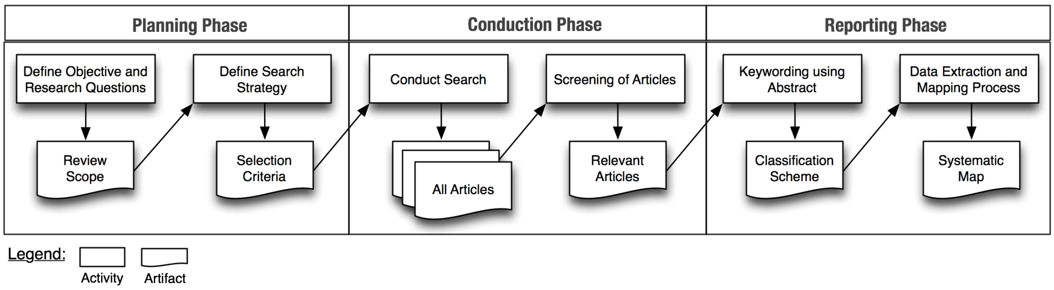

Fig. 1: The systematic mapping process used in our study (adapted from [20]).

planning phase (Section 2), conduction phase (Section 3) and reporting phase (Section 4 and Section 5). This protocol addresses steps, claimed as essential in [17, 19, 20], to plan and create a systematic mapping of the literature. Moreover, our study protocol is based on previously validated systematic mapping studies [11, 60–62]. This article makes use of systematic mapping study as methodology because it minimizes bias when compared to single literature reviews, thus getting more reliable findings [22]. Mapping study does not only discuss the findings obtained, but also seeks to properly describe all activities required to run the review as a whole.

2.1 Objective and research questions

The objective of this work is to provide an overview of the literature by classifying currently available articles, pinpointing gaps, and promising research directions for further investigation. In addition, we perform a meta-analysis by presenting a quantitative synthesis of results produced from different studies. Lau et al. [51] emphasize that this meta-analysis by combining information from "different studies can increase statistical power and provide answers that no single study can give" [51]. For this, we seek to analyze the literature concerning the following variables: cost-drivers that were used more frequently, research methods used to evaluate the studies, research issues addressed, participants, quality assessment and research venue. With this in mind, we state the objective of our study based on the GQM template [49] as follows:

Analyze *the current literature*
for the purpose of *investigating its state*
with respect to *used cost-drivers, research methods, research issues, participants, quality assessment, and research venue*
from the perspective of *researchers and practitioners*
in the context of *effort estimation.*

To properly address this objective, we define six research questions (RQs), which are presented in Table 1, along with their motivations and explored variables. Petersen et al. [19] mention that the RQs of systematic mapping studies should be generic so that research trends over time, and topics covered in the literature can be identified. For this reason, these RQs are generic so that a literature overview can be created.

2.2 Search strategy

After defining the research questions, the next step was to determine a search strategy. For this, we followed the well-known empirical guidelines discussed in [14, 17, 19, 23] so that an unbiased and iterative search strategy could be elaborated. Our focus was the definition of search strings, which were fundamental to select a list of representative studies of the current literature. Table 2 shows the key terms and alternative terms of the search strings. These terms were determined after a careful review of the most commonly used keywords in the selected articles.

Steps to define our search strings. We adopted the following aspects to define our search string:

1. Specify the main keywords;
2. Define alternative words, synonyms or related terms to chief keywords;
3. Check if the major keywords are contained in the articles;
4. Associated synonyms, alternative words or terms related to the main keywords with the Boolean "OR"; and
5. Link the major terms with Boolean "AND".

Several combinations of such search terms were formulated and applied to four electronic databases, which are listed in Table 3. The combinations that produced the most significant results are presented as follows:

(Software OR Application OR Product) AND
(Effort OR Cost OR Pricing OR Time) AND
(Estimation OR Evaluation OR Measurement OR Judgment) AND
(Development OR Maintenance OR Evolution) AND
(Prediction OR Effort) AND
(Project OR Productivity)

The next step was to define where the current literature would be retrieved (i.e., the source of information). Table 3 shows four electronic databases that were used to retrieve works. These databases were chosen because of their elevated number of studies stored. Moreover, they have been widely used in previous systematic mapping studies [10, 60–62].

2.3 Exclusion and inclusion criteria

This section aims to establish criteria to exclude and include candidate studies retrieved from the selected electronic databases showed in Table 3. The following Exclusion Criteria (EC) were used to discard works that:

- **EC1:** The title, abstract or even their content was not closely related to our search string, however without any semantic interplay;
- **EC2:** Were not published in English, were patent, or might be considered as an initial stage, typically represented by abstract and summary;
- **EC3:** No similarity with the research theme, or even the focal aim was completely contrary to the purpose of the issues addressed in the research questions;
- **EC4:** No aspect of the research questions was found in the abstract;
- **EC5:** It was a duplicate; and
- **EC6:** It did not address issues about effort estimation.

Reasons for choosing the exclusion criteria. These exclusion criteria were chosen because some reasons. First, it would not make any sense to consider studies without any semantic relation with the subject of effort estimation, just because the search string has matched with its title or abstract (EC1). Second, early-stage studies would contribute not so much to create an overview of the area, as well as generate important findings in this study. On the contrary, they could misrepresent the creation of an overview of the literature, the definition of trends and gaps (EC2). Third, we believe that work

Table 1 Research questions that were investigated in this article.

Research Question	Motivation	Variable
RQ1: Which cost-drivers have been most commonly used?	Determine which the main concepts are, study models and researches applied in the software industry, and survey the state-of-the-art approaches.	Used cost-drivers
RQ2: What are the most commonly used research methods?	Reveal which research methods have been most commonly used in practice.	Research method
RQ3: Which research issues have been investigated more frequently?	Reveal which issues have been explored in the last decades. considering effort estimation.	Research issue
RQ4: Who participates in the studies?	Understand who often participates in studies about effort estimation.	Study participant
RQ5: How can the current studies be qualitatively assessed?	Evaluate the literature concerning qualitative issues.	Quality assessment
RQ6: Where have the studies been published?	Elicit the target venues used to disclose the results.	Research venue

Table 2 A description of the major terms and their synonyms.

Main Term	Alternative Terms
Software	Application, Product, Project
Effort	Cost, Pricing, Time
Estimation	Evaluation, Measurement, Judgment
Development	Maintenance, Evolution, Productivity

Table 3 List of the selected electronic databases.

Source	Electronic Address
ACM DL	dl.acm.org
IEEE Xplore	ieeexplore.ieee.org/Xplore/home.jsp
Science Direct	www.sciencedirect.com
Scopus	www.scopus.com

that was not minimally related to research questions could not contribute to effectively answer the research questions explored (EC3). Fourth, if the abstract of an article did not present any aspect of investigations explored by the research questions, then it would not make sense to consider it (EC4). Fifth, it would not make sense to consider investigating duplicate studies (EC5), or not addressing the subject of effort estimation (EC6). More importantly, these exclusion criteria have already been validated in previous studies, such as [10, 21, 50, 60].

Application of the exclusion criteria. The exclusion criteria were applied by the authors manually, since this application requires the understanding of the surveyed articles as a whole. The process of applying the criteria was performed following an iterative and incremental process. The exclusion of an article was always based on the authors' consensus, so that bias might be minimized. To this end, two review cycles were performed to avoid any unwanted removal of articles. Moreover, we describe the adopted criteria to include the candidate works in our sample, which were retrieved from the electronic databases, into our list of selected studies (Section 3). The Inclusion Criteria (IC) are presented as follows:

- **IC1:** Academic works (i.e., articles, surveys, papers, master and doctoral thesis) aimed to propose cost-drivers, report empirical results or survey;
- **IC2:** Works written, published or disseminated in English;
- **IC3:** Works found in scientific journals, conferences, research groups'web page or educational institutions; and
- **IC4:** Studies published from January 2000 until December 2016.

2.4 Data extraction

This section explains how we extracted data from the selected studies, which are presented in Section 3. The data extraction procedures consist of reading each selected study carefully and storing the extracted data in a spreadsheet. For this, the data extraction form, shown in Figure 2, was used. This form is based on well-validated one found in [21]. It served as a template for easing the data synthesis, enabled us to carefully obtain data and generate qualitative indicators, as well as plot evidence about the formulated research questions (Table 1). The data extraction generates numerical values, nominal or ordinal data, which were crucial for any attempt to create a snapshot of the current literature. Table 4 presents the classification scheme used to extract data from the selected works.

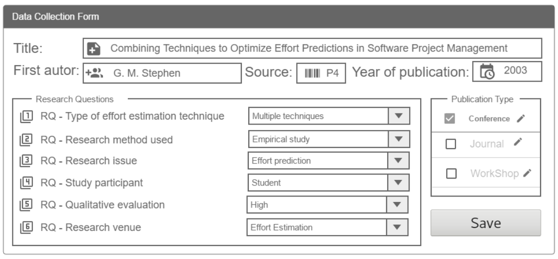

Fig. 2: An illustrative form to extract data from the selected studies.

In RQ1, each study was reviewed and classified as multiple cost-drivers, COCOMO, use case point, story point, function point or other. In consideration of the research method used (RQ2), we used an existing classification of research approaches, proposed by Wieringa et al. [57]. Recent studies (e.g., [19, 58]) also used this classification scheme for the same purpose. Our research classification scheme considers the following research methods (based on [19]):

- **Evaluation research:** Studies sketching a particular problem, proposing a solution and conducting an empirical analysis, so that the benefits and drawbacks can be drawn.
- **Philosophical papers:** Studies looking for proposing a taxonomy or conceptual framework as a way of sketching a new way of looking at existing approaches about cost-drivers.
- **Experience papers:** Studies reporting the personal perception of the author on the use of a particular cost-driver. Usually, these studies explain on what and how something has been done in practice.
- **Opinion papers:** Studies reporting the personal opinion of somebody regarding the quality of cost-drivers. This evaluation does not take into account related works or research methodologies.
- **Solution proposal:** Studies proposing a solution for a particular problem, being the solution either novel or a significant extension of previous ones. Petersen et al. [19] highlight small examples are typically used to demonstrate the potential benefits and the applicability of the proposed solution.
- **Validation research:** Studies explored are novel, but have not yet been implemented and used in production. These studies refer, for example, experiments, i.e., work done in the lab [19], to test prototypes. Usually, it evaluates an early sample, model, or release of a technique, and serves as a first step to measure the proposed technique, and acts as an initiative to be replicated or learned from.

The RQ3 addresses research issues explored in the selected studies, including effort estimation, effort prediction, cost prediction, expert judgment, empirical studies, and cost estimation. The term cost estimation refers to the usage of empirical models based on experience, represented as historical results in a database, or mathematical formulas to calculate the effort to be employed in the execution of a development task. On the other hand, the term cost prediction refers to the usage of mathematical models to exclusively achieve an effort measure. Note that in cost prediction, experience is not taken into account. However, the values of environmental variables can be considered in the prediction, and if they are changed, they may present different results. The RQ4 tries to reveal who are the participants, considering two categories students and professionals. The RQ5 classifies the studies in terms of their quality, such as high, medium and low. We explain how this classification is computed in Section 2.5.

Table 4 The used classification scheme to extract data.

Research Question	Variable	Answers
RQ1	Used cost-drivers	Multiple cost-drivers, COCOMO, use case point, function point, story point or other
RQ2	Research method used	Evaluation research, philosophical papers, experience papers, opinion papers, solution proposal, validation research
RQ3	Research issue	Effort estimation, effort prediction, cost prediction, expert judgment, empirical studies, cost estimation, function point
RQ4	Study participant	Student, professional
RQ5	Quality assessment	High, medium, low

2.5 Quality assessment

The form shown in Table 5 was elaborated to qualitatively assess the selected studies. This assessment form enables us to examine some key quality issues, thereby classifying the selected studies in terms of quality. In total, six quality issues are determined. Each one has a specific purpose and score, as follows: (*i*) aspects and objectives (4 points); (*ii*) context definition (2 points); (*iii*) experimental design (2 points); (*iv*) data analysis techniques (4 points); (*v*) threats to validity (4 points); and (*vi*) conclusions (4 points). The studies that counted less than 10 points, from 10 to 15 points, and more than 15 points were classified as low, medium and high quality, respectively.

This quality assessment form was elaborated based on previously ones, available in [21, 24, 25, 27–30]. The entries of the form can be justified by explaining the reasons to explore these issues. The first issue seeks to understand the objectives and definition of scope. For this, we evaluate if the research objectives are clearly defined and motivated. Additionally, we check if the research questions, hypotheses or theories are properly stated.

Second, it explains about the context definition. We are concerned on investigating if the context in which the explored studies were run is properly delineated, and the experimental procedures are carefully stated. Third, this issue addresses the experimental design. We evaluate whether the experimental design is aligned with the study objective. In addition, we check if researches participated in the study. If so, we verify how they were allocated to experimental groups. Note that if experimental design is not properly addressed, then all planning of a study to meet specified objectives can be compromised. Fourth, we investigate if the study describes the data analysis procedures adopted, and reports significance levels, effect sizes and power of statistical tests applied.

Furthermore, we examine if a statistical method used is proper to the collected data quality. Fifth, we also concern on checking if threats to validity of the collected data were identified and mitigated appropriately. In our last issue, we assess if the results are clearly demonstrated, if conclusions are also properly stated and supported by the collected data. Moreover, we analyze if the study authors discuss the interplay between the research questions and findings.

3 Study Filtering

This section describes how the conduction phase (Figure 1) of our study was performed. For this, we introduce the process applied to filter candidate studies. This filtering is formed by nine steps, in which the exclusion criteria are applied. Figure 3 illustrates the results obtained in each step. Each step of the study filtering process is described as follows:

- **Step 1: Initial search.** Bring together the initial results, after submitting the search string to the electronic databases (Table 3). In total, 3,746 candidate studies were retrieved.
- **Step 2: Impurity removal (EC1 & EC2).** We then applied two exclusion criteria, EC1 and EC2, to remove impurities. After applying EC1 and EC2 (described in Section 2.3) to the primary studies, some were thrown away due to the absence of any semantic interplay of their title, abstract or even contents with regards to the subject investigated in our article (i.e., out of scope). In addition, studies that were not written in English were also discarded. In total, 1,263 studies (33.72%) remained for the next step while 66.28% (i.e., 2,483 works) were removed. Examples of these works that were retrieved would be call for papers of conferences, special issues of journals, patent specifications, research reports and no peer reviewed materials.
- **Step 3: Filter by similarity (EC3).** This step removed all studies that did not have similarity with our search string. For this, 72.05% (910 of 1,263) of the studies were filtered.
- **Step 4: Filter by abstract (EC4).** This step examined 910 studies based on their abstract. In total, 79.67% (725 of 910) of the articles were filtered. It was possible to remove studies whose content was not closely related to the key issues addressed by our research questions.
- **Step 5: Combination.** The remaining studies were brought together to produce a sample of 725 candidate studies.
- **Step 6: Duplicate removal (EC5).** Usually, a study can be found in several digital libraries. Thus, we applied EC5 to remove all duplicates, thereby ensuring the uniqueness of each study: 48.68% (353 of 725) were filtered.
- **Step 7: Study addition by heuristic.** Although the search mechanisms (Table 3) are widely recognized, some works may not be retrieved. Thus, we inserted certain studies manually to our primary studies to mitigate this threat. We added 21 studies by applying heuristics and a snowballing process [47], producing a sample of 374 studies (i.e., 5.95% added). We reviewed the DBLP of some authors, and the references and citations of the articles themselves.
- **Step 8: Filter by full text (EC6).** After reading the full text of the remaining 374 studies, 33.42% were filtered by applying the EC6, excluding studies whose the contents were away from the expected issues on effort estimation and closely related to our RQs. The following rules were applied to support our filtering process:
  - *Rule 1:* Articles whose content was related to the theme of effort estimation, but was not applied to the area of software engineering. Although they have been identified by our search string, their content is not within the scope of this work.
  - *Rule 2:* Articles of literature review were removed and tried as related works.
  - *Rule 3:* Articles whose size was small (up to 2 pages) were also filtered.
  - *Rule 4:* Articles that were not directly aligned with the purpose of our article (Section 2.1). That is, we filtered all articles that were met by our search strings, but their content was not closely related to the purpose of this article.
- **Step 9: Representative work selection.** By exploring the remaining 125 studies, we observed that some were technically similar, i.e., studies produced based on previous ones, and their contributions were closely related. Thus, 96% were selected. Finally, 120

IET Research Journals, pp. 1–14

Table 5 The quality assessment form (based on [21, 24–26])

Criterion	Quality (Max: 20 points)
1. Aspects and Objectives	Total: 4 points
1.1. Are the research objectives clearly defined?	If the objectives are defined (Add 1 point)
1.2. Is the study properly motivated?	If the reasons are properly explained (Add 1 point)
1.3. Are the research questions defined?	If the research questions are defined (Add 1 point)
1.4. Does the study define hypotheses and theories?	If the hypotheses are presented and explained (Add 1 point)
2. Context Definition	Total: 2 points
2.1. Does the study describe the sample and experimental steps?	If the sample and experimental steps are properly presented (Add 1 point)
2.2. Does the study explain the context?	If the experimental context is outlined (Add 1 point)
3. Experimental Design	Total: 2 points
3.1. Do the authors explain the design research?	If the experimental design is explained (Add 1 point)
3.2. Do the authors define and describe all treatments?	If the treatments are described (Add 1 point)
4. Data Analysis Techniques	Total: 4 points
4.1. Does the study explain their choices and describe the data analysis procedures?	If the choices and data analysis procedures are described (Add 1 point)
4.2 Does the study report significance levels, effect sizes and power of tests?	If there is a significant explanation (Add 1 point)
4.3. Does the study have sufficient data to execute the validation process?	If there is the sufficient data to analyze (Add 1 point)
4.4. Does the statistical test provide support for the assessment to the collected data quality?	If the descriptive statistics are reported (Add 1 point)
5. Threats to Validity	Total: 4 points
5.1. Is the relationship between researchers and participants with the experiment execution considered?	If the relationship is considered (Add 1 point)
5.2. Do the authors explain the implications results over the users?	If the implications are explained (Add 1 point)
5.3. Do the participants receive training properly?	If the training is presented (Add 1 point)
5.4. The threats to validity were discussed by the authors?	If the threats to validity are presented (Add 1 point)
6. Conclusions	Total: 4 points
6.1. Do the authors present results clearly?	If the results are properly presented (Add 1 point)
6.2. Do the authors present conclusions clearly?	If the study concludes and indicates new research avenues (Add 1 point)
6.3. Are the conclusions supported by the collected data?	If the conclusions are extracted from data results (Add 1 point)
6.4. Do the authors discuss the interplay between the research questions and the results?	If there is interplay between research questions and conclusions (Add 1 point)

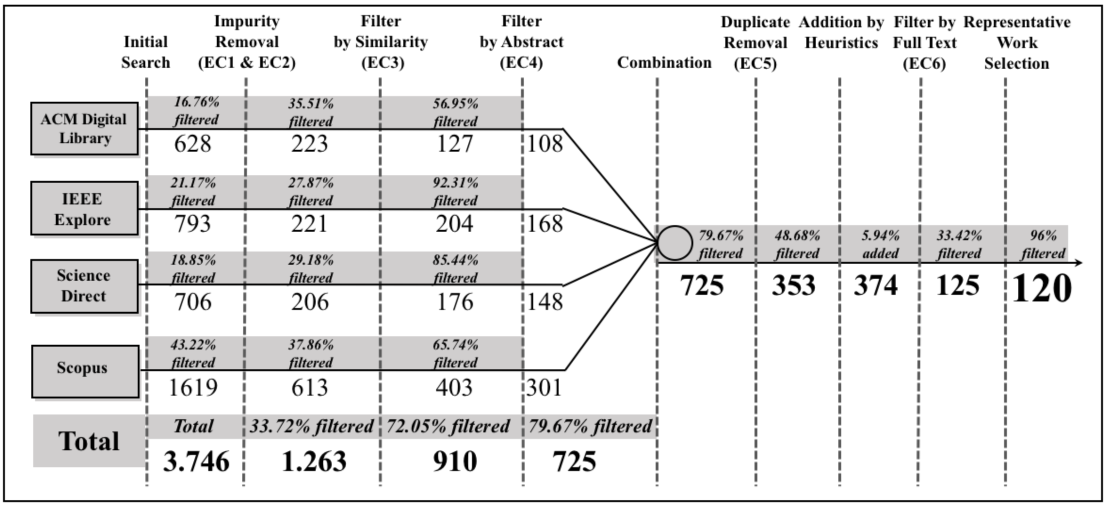

Fig. 3: The selected studies throughout the filtering process.

works were selected as the most representative ones, hereinafter called *primary studies* (presented in Appendix A).

Figure 4 shows the stratification of the obtained results from the search. To mitigate issues related the reliability of the filtering process, three review cycles were performed to discuss quality issues and avoid false positives/negatives, which might appear due to improper application of the inclusion/exclusion criteria. The authors were fully involved in each filtering step, which lasted six months so that the articles might be selected and classified, two months to extract data, and by about five months for writing and reviewing this mapping study.

Furthermore, the authors were always available to discuss the correct application of the selection and evaluation criteria. Therefore, we believe that all these aspects provided a careful selection process, contributing to create a common sense about the quality, and form representative situations where all authors might inquire the quality of the candidate articles up front. Table 13 (in Appendix B) shows the final punctuation of the primary studies evaluated in our work.

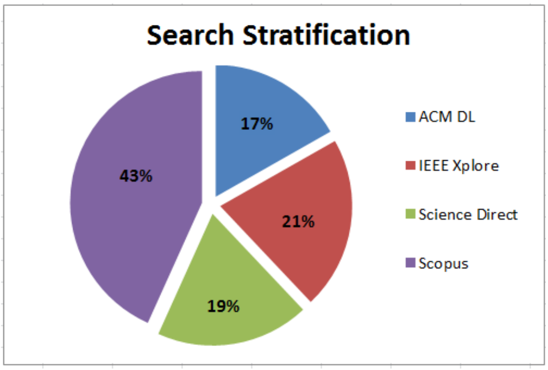

Fig. 4: Search stratification based on the electronic databases used.

4 Study results

4.1 RQ1: Which cost-drivers have been most commonly used?

The RQ1 aims to reveal the cost-drivers that have been often used. Table 6 shows the results related to the RQ1. The main feature is that most studies (71.67%, 86/120) use multiple cost-drivers rather than priorite a specific one. Examples of these cost-drivers would be: COCOMO (Constructive Cost Model), effort correlation, function point ([P120, P119]) and use case point (e.g., [P25, P26], [8, 24, 31–36]). The collected data also suggest that COCOMO (Constructive Cost Model) is the second most-used cost-driver (11.66%, 14/120). A lower amount of primary studies (2.50%, 3/120) has applied use case or story point to estimate development effort and (1.67%, 2/120) primary studies used function point. Additionally, we highlight that some studies (12.5%, 15/120) aim to apply statistical methods to predict effort, including regression multivariate analysis [37], evolutionary computing [38–41], Bayesian Belief Network [42], error rate variation, fuzzy logic and linear regression [43, 44]. To address the problem of estimating effort based on a particular cost-driver, multimodal estimation mechanisms might be proposed in upcoming studies to eliminate the weakness of relying on one data source or information in unimodal estimation approach.

Table 6 The cost-drivers that have been most commonly used (RQ1).

Answers	Amount	Percentage	List of primary studies
Multiple cost-drivers	86	71.67%	[P02], [P04], [P05], [P06], [P07], [P08], [P10], [P11], [P12], [P13], [P14], [P15], [P16], [P17], [P18], [P19], [P21], [P22], [P23], [P25], [P27], [P28], [P29], [P30], [P34], [P36], [P37], [P39], [P40], [P41], [P43], [P45], [P50], [P51], [P52], [P53], [P54], [P55], [P56], [P58], [P59], [P64], [P65], [P69], [P70], [P72], [P73], [P74], [P76], [P78], [P79], [P82], [P83], [P84], [P85], [P86], [P87], [P89], [P90], [P91] [P92], [P93], [P94], [P95], [P96] [P97], [P98], [P100], [P101], [P102], [P103], [P104], [P105], [P106], [P107], [P108], [P109], [P110], [P111], [P112], [P113], [P114], [P115], [P116], [P117], [P118]
COCOMO	14	11.66%	[P20], [P24], [P26], [P31], [P32], [P33], [P38], [P44], [P46], [P47], [P48], [P49], [P60], [P66]
Use case or story point	3	2.50%	[P35], [P88], [P99]
Function point	2	1.67%	[P119], [P120]
Others	15	12.50%	[P01], [P03], [P09], [P42], [P57], [P61], [P62], [P63], [P67], [P68], [P71], [P75], [P77], [P80], [P81]

4.2 RQ2: What are the most commonly used research methods?

The RQ2 seeks to reveal the most commonly used research methods in the primary studies. For this, the classification scheme, shown in Table 4, was used, and Table 7 shows the obtained data. In total, most primary studies (45.83%, 55/120) focused on performing evaluation research, while 19.17% (23/120) produced philosophical papers. Together, experience and opinion papers computed a total of (24.17%, 29/120). While experience papers report authors' personal experience about how specific practices have taken in practice, opinion papers do not rely on related work and research methodologies. Finally, solution proposal and validation research received little attention, registering (7.50%, 9/120) and (3.33%, 4/120) respectively.

Although some works performed evaluation research, the experimental design of the studies were not similar. This means that these studies, and their results, may not be comparable. Therefore, their collected results cannot be generalizable. It would be very important if a case study could be performed in several companies. This replication of the study in different contexts will generate results with a greater degree of confidence. An insightful recommendation to the community would be to produce new studies — or even previous ones — based on the methods already used in the primary studies so that the results can be comparable. This replication of the study in different contexts will generate results with a greater degree of confidence.

Table 7 The research method used by primary studies (RQ2).

Answers	Amount	Percentage	List of primary studies
Evaluation research	55	45.83%	[P01], [P02], [P11], [P12], [P15], [P19], [P20], [P25], [P27], [P28], [P26], [P37], [P34], [P43], [P52], [P53], [P54], [P55], [P56], [P59], [P62], [P64], [P68], [P69], [P71], [P72], [P75], [P78], [P82], [P83], [P84], [P85], [P86], [P87], [P88] [P92], [P93], [P94], [P95], [P96] [P97], [P98], [P99], [P100], [P101] [P102], [P103], [P105], [P106], [P108], [P109], [P110], [P114], [P119], [P120]
Philosophical papers	23	19.17%	[P17], [P29], [P30], [P31], [P32], [P38], [P42], [P47], [P48], [P57], [P58], [P73], [P76], [P77], [P80], [P81], [P89], [P90], [P104], [P107], [P111], [P117], [P118]
Experience papers	17	14.17%	[P09], [P13], [P24], [P35], [P36], [P40], [P41], [P44], [P49], [P61], [P66], [P67], [P74], [P112], [P113], [P115], [P116],
Opinion papers	12	10%	[P10], [P14], [P18], [P23], [P33], [P39], [P45], [P51], [P60], [P63], [P65], [P70],
Solution proposal	9	7.50%	[P04], [P05], [P06], [P07], [P21], [P22], [P46], [P79], [P91]
Validation research	4	3.33%	[P03], [P08], [P16], [P50]

4.3 RQ3: Which research issues have been investigated more frequently?

The RQ3 focuses on investigating the research topics that have been most explored in the current literature. Table 8 depicts the obtained data regarding the RQ3. Six research issues were explored. First, most primary studies focused on research topics related to effort estimation (68.33%, 82/120). Effort prediction is the second most explored issue (10.84%, 13/120). This category explores studies that use cost-drivers to predict effort to be invested in software development tasks. For example, Briand et al. [45] pointed out that the ordinary least-squares regression presented the best results, compared to the other approaches. Expert judgment and cost prediction registered 5.83% (7/120) and 4.16% (5/120) respectively. Finally, empirical studies (3.34%, 4/120), cost estimation (5.83%, 7/120) and function point (1.67%, 2/120) registered four studies for each ones, respectively.

We identified that seven research topics were commonly explored (effort estimation, effort prediction, cost prediction, expert judgment, and empirical studies). This means that the estimation of project's effort through approaches still remains a constant concern in software engineering community, being explored through various perspectives. We also perceived that although the subject of effort estimation has been widely explored in recent decades, it still remains widely current and required by industry. The need for accurate effort estimations for agile projects, for example, is one of the most critical issues in the software industry, where software requirements are increasingly volatile.

Moreover, an in-depth analysis of results about empirical studies like case studies might also be done. Typically, case studies are run to explore characteristics (e.g., profiles of development team) or phenomena (e.g., a fact observed in agile practices) related to effort estimation in its real-life context in a specific time space. Such estimation-influencing phenomena and characteristics may be hard to clearly distinguish from its environment. This means that the produced empirical results ending up being hard to be applied in other contexts, making it difficult to produce accurate estimates.

The primary studies classified as *effort prediction* used predictors, with a set of data, to guess at some random value that is not part of an initial dataset. As prediction is a part of statistical analysis, some statistical techniques were perceived in our primary studies, including regression analysis and its facets, such as linear regression, logistic regression, and Poisson regression. In general, these studies, supported by predictive inference, sought to generate knowledge from previous projects, so that new estimates might be produced. The primary studies classified as *effort estimation* made use of estimators, along with data, to guess at a parameter. These studies also had knowledgeable persons in a particular area generated estimates based on their experience, or even inductive and deductive reasoning. For example, Brown [7] proposed the Delphi method as an approach for defining expert-judgment-based predictions in a controlled way. Thus, an interesting research direction would be to explore a matching between intuitive probability curves generated by experts, and prediction elaborated from statistical inference.

Table 8 The investigated research topics (RQ3).

Answers	Amount	Percentage	List of primary studies
Effort estimation	82	68.33%	[P01], [P02], [P03], [P05], [P07], [P09], [P10], [P11], [P12], [P13], [P14], [P15], [P17], [P18], [P19], [P20], [P21], [P22], [P23], [P24], [P27], [P28], [P30], [P31], [P32], [P33], [P36], [P37], [P38], [P39], [P40], [P41], [P43], [P44], [P45], [P49], [P50], [P51], [P57], [P58], [P60], [P61], [P64], [P67], [P69], [P70], [P72], [P74], [P76], [P77], [P78], [P79], [P82], [P83], [P85], [P86], [P87], [P88], [P89], [P91], [P92], [P93], [P94], [P95], [P96], [P98], [P99], [P100], [P101], [P102], [P103], [P104], [P106], [P107], [P108], [P110], [P112], [P113], [P115], [P116], [P117], [P118]
Effort prediction	13	10.84%	[P04], [P08], [P53], [P55], [P59], [P65], [P68], [P75], [P80], [P81], [P84], [P90], [P97]
Cost Prediction	5	4.16%	[P06], [P26], [P29], [P47], [P71]
Expert judgment	7	5.83%	[P34], [P42], [P46], [P48], [P52], [P63], [P54],
Empirical studies	4	3.34%	[P16], [P25], [P35], [P62],
Cost Estimation	7	5.83%	[P56], [P66], [P73], [P105], [P109], [P111], [P114]
Function Point	2	1.67%	[P119], [P120],

4.4 RQ4: Who participates in the studies?

The RQ4 aims to investigate who has participated in the primary studies. Table 9 shows the data related to the RQ4. The main result is that students (or researchers) are the most common participants in the primary studies (91.67%, 110/120), while practitioners participate in a reduced number of studies (8.33%, 10/120). Although researchers recognize the value of students in empirical studies [46], the presence of professionals in effort estimation studies is critical to produce realistic findings. If results are produced considering students only, then it is questionable to apply such empirical knowledge in practice where time is tight and is predominantly formed by professionals, rather than beginners.

Table 9 Participants of the primary studies (RQ4).

Answers	Amount	Percentage	List of primary studies
Student	110	91.67%	[P01], [P02], [P03], [P04], [P05], [P06], [P07], [P08], [P09], [P10], [P11], [P12], [P13], [P14], [P15], [P16], [P17], [P18], [P19], [P21], [P22], [P23], [P24], [P25], [P26], [P27], [P28], [P29], [P30], [P31], [P32], [P34], [P37], [P39], [P40], [P41], [P42], [P43], [P44], [P45], [P47], [P48], [P49], [P50], [P51], [P52], [P53], [P54], [P55], [P56], [P57], [P58], [P59], [P60], [P61], [P62], [P63], [P64], [P65], [P67], [P68], [P69], [P70], [P71], [P72], [P73], [P74], [P75], [P76], [P77], [P78], [P79], [P80], [P81], [P82], [P83], [P84], [P85], [P86], [P87], [P88], [P89], [P90], [P91], [P92], [P93], [P94], [P95], [P96], [P97], [P98], [P99], [P100], [P101], [P102], [P103], [P104], [P105], [P106], [P107], [P108], [P109], [P111], [P114], [P115], [P116], [P117], [P118], [P119], [P120]
Professional	10	8.33%	[P20], [P33], [P35], [P36], [P38], [P46], [P66]. [P110], [P112], [P113]

4.5 RQ5: How can the current studies be qualitatively assessed?

The RQ5 seeks to assess the primary studies in terms of quality. For this, we elaborated a quality assessment form present in Table 5. Each study was evaluated according to the issues presented in this form. As previously cited in Section 2.5, we used this quality assessment form because it has been validated in previous studies, such as [21, 24–26]. Table 10 presents the obtained results regarding the quality level of the primary studies. In part, this result shows the maturity of the research area by presenting a quantitative synthesis of the obtained results after aggregating information from the different studies. Lau et al. [51] argue that this quantitative synthesis, named as meta-analysis, can increase statistical power, as well as supply answers that no single study can give [51].

The main finding is that most primary studies presented high quality (62.5%, 75/120), while a minor part presented a medium quality (33.33%, 40/120) or a low quality (4.17%, 5/120). The studies that counted less than 10 points, from 10 to 15 points, and more than 15 points were classified as low, medium and high quality, respectively. While Table 10 shows the overall results of qualitative analysis, Table 13 (in Appendix B) presents all data.

4.6 RQ6: Where have the current studies been published?

This section seeks to explore the major point of the RQ6, i.e., reveal where the primary studies are being published over the years. We classified the primary studies based on the publication year, type of publication (i.e., journal, conference and workshop papers) and the amount of studies published by year. Figure 5 presents the data

Table 10 Classification of the primary studies (RQ5).

Answers	Amount	Percentage	List of primary studies
High	75	62.5%	[P01], [P02], [P03], [P04], [P05], [P08], [P09], [P10], [P11], [P12], [P13], [P14], [P16], [P17], [P18], [P20], [P23], [P24], [P25], [P28], [P29], [P30], [P31], [P32], [P37], [P39], [P43], [P49], [P51], [P54], [P56], [P57], [P58], [P61], [P62], [P64], [P67], [P68], [P69], [P70], [P71], [P72], [P73], [P74], [P75], [P76], [P77], [P80], [P81], [P82], [P83], [P84], [P87], [P88], [P89], [P90], [P91], [P92], [P93], [P94], [P95], [P96], [P99], [P100], [P101], [P102], [P103], [P104], [P105], [P107], [P108], [P111], [P116], [P119], [P120]
Medium	40	33.33%	[P06], [P07], [P15], [P19], [P21], [P22], [P27], [P33], [P34], [P35], [P36], [P38], [P41], [P42], [P44], [P45], [P46], [P47], [P50], [P52], [P53], [P55], [P60], [P63], [P65], [P66], [P78], [P79], [P85], [P86], [P97], [P98], [P106], [P109], [P112], [P113], [P114], [P115], [P117], [P118]
Low	5	4.17%	[P26], [P40], [P48], [P59], [P110]

Table 11 Amount of publications by author.

Author	Quantity	Percentage
Emilia Mendes	20/120	16.67%
Magne Jorgensen	17/120	14.16%
Ali Idri	14/120	11.66%
Alain Abran	14/120	11.66%
Ricardo Britto	9/120	7.50%
Tayana Conte	4/120	3.33%
Martin J. Shepperd	3/120	2.50%
Stein Grimstad	3/120	2.50%
Kjetil Molokken-Ostvold	3/120	2.50%

obtained on these points. This distribution of the primary studies over the years helps to create a panoramic view of the current literature in terms of time. We noted that, although effort estimation is not a recent area of research, the number of published articles continues to grow. This result demonstrates that this area of research is very active.

Table 12 presents the main research venue where the primary studies were published. We can note that the Journal of Systems and Software is the vehicle where the articles have been most published (13.34%, 16/120). The Journal Information and Software Technology registered (5.83% 7/120), followed by IEEE Software with (4.17%, 5/120). The collected data suggest that there is no predominant vehicle in which researchers have prioritized the publication of their articles. On the contrary, we observed a great heterogeneity in relation to the place of publication.

5 Discussion and future directions

This section presents discussions about the collected data by investigating which are the most explored research topics over the years, and pinpointing who are the researchers who have most published articles. Moreover, we introduce some future directions about the use of cognitive load, and domain, human and team factors in effort estimation.

Most explored research topics. We also seek to understand which research topics are being explored and how the primary studies have been published over the years. For this, we organize our primary studies through three dimensions of their data using a Bubble chart shown in Figure 6. Each Bubble has a triplet (*t1*, *t2*, *t3*) of obtained data, where *t1* represents the research topic explored, *t2* is the publication year, and *t3* consists of the amount of studies exploring a particular research topic. This chart enables us to understand, based on quantitative evidence, in which research topics the community has put more attention. The results suggest that effort estimation is the area most explored over the years, while works aimed at empirical studies are still scarce.

Most active researchers. We seek to identify the researchers who have most contributed to the field of effort estimation in the last years. For this, Table 11 shows the number of publications by author. Emilia Mendes and Magne Jorgensen has been the most active researchers with 20 and 17 publications. Ali Idri and Alain Abran were the third most active researcher with 14 publications.

Cognitive Load. Recent studies have used the measure of cognitive load as a base to calculate the performance of developers on software-development tasks [52, 54]. According to Fritz and Muller, cognitive load refers to the cognitive effort that developers should apply to perform a development task [52, 53]. Today, we have learned from previous studies [52, 53, 55, 56] that the effort that developers invest to do a development task can be narrowly related to the cognitive load realized by them. This means that the higher measures of cognitive effort, the lower performance of developers [54]. Unfortunately, the studies fail short to usage the cognitive load of developers to estimate effort; rather, the current works have neglected the use of cognitive data to predict temporal effort. Future works might investigate the relation of the developers' cognitive load and the required temporal effort to properly carry out development tasks. For example, linear regression modeling might be used to explore the overall associations between cognition factors and types of development tasks.

In addition, identifying human, organizational and technical factors that are related to increased cognitive load would be another relevant work. We conjecture that the level of artifact abstraction, the degree of task complexity, organizational culture, and time constraints can influence cognitive activities and, consequently, productivity. However, there is no empirical evidence that actually tests this hypothesis; it is not even known to what extent such factors could influence estimates if this hypothesis was confirmed. Moreover, the development of innovative estimation approaches for recommending tasks according to their cognitive load would also be a significant contribution. Specifically, smart estimation approaches might be created to recommend development tasks (e.g., testing, coding and modeling) based on machine learning techniques, such as deep learning. These new approaches could recommend complex tasks for people who are likely to have to spend a low cognitive load to do those tasks, thereby reducing the likelihood that estimates will not be met.

Domain, human and team factors matter to effort estimates. The team definition is a pivotal task in software development as teams should properly interact with each other to achieve certain levels of productivity. If collaborative teams are not created, then productivity-favoring situations cannot often be experienced. Hence, effort estimation approaches tend not to be effective; no matter how good they are. This means that effective productivity situations need to be properly sized and considered when estimating development costs, which are related to factors not previously considered in the current literature, such as domain, human and team factors. When we try to understand how such factors together could influence the practice of estimating effort, no reasonable line of reasoning can be proposed or even found in the literature.

In this sense, a promising research direction would be to propose a prediction model that could determine estimates by (i) considering the configurations and profiles of software-development teams created, as well as (ii) suggesting team configurations that could achieve a greater number of high-quality interactions. Still, none of the primary studies considered the combination of domain, human and team factors — e.g., application domains, implementation platforms, emotional states, gender, individual and group interest, development methodologies, cultural issues, and interaction patterns — and their relationships with effective estimates. Therefore, we believe that domain, human, and team factors are essential for developing more realistic effort estimates.

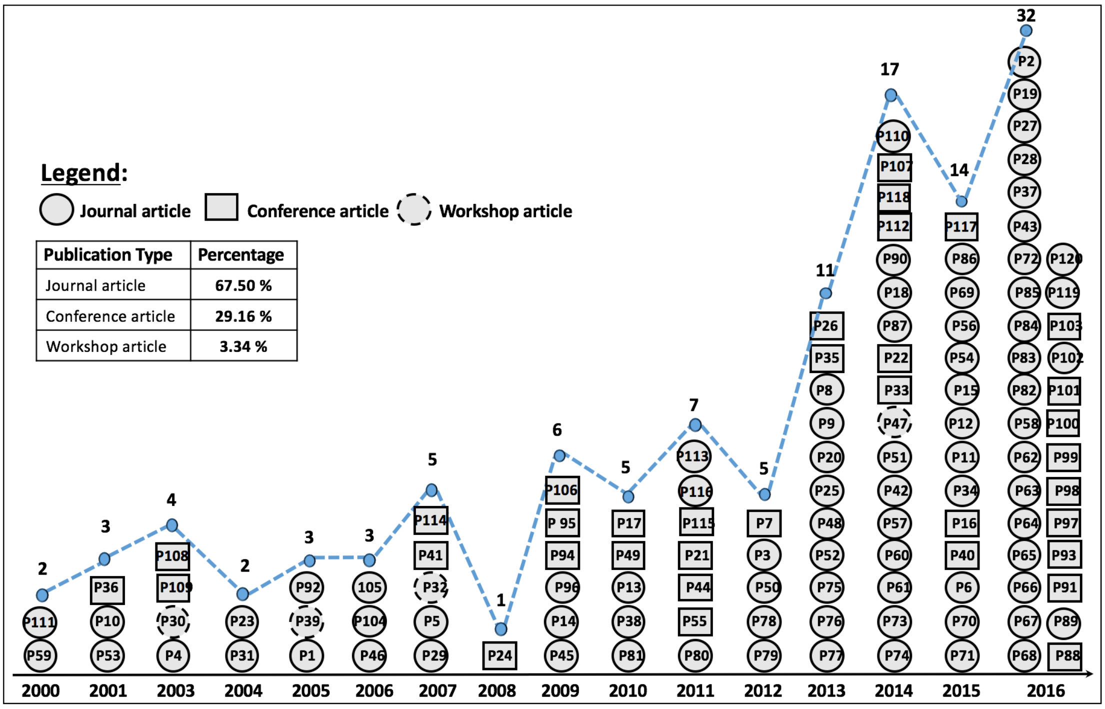

Fig. 5: Distribution of the primary studies over the years (based on [13]) (RQ6).

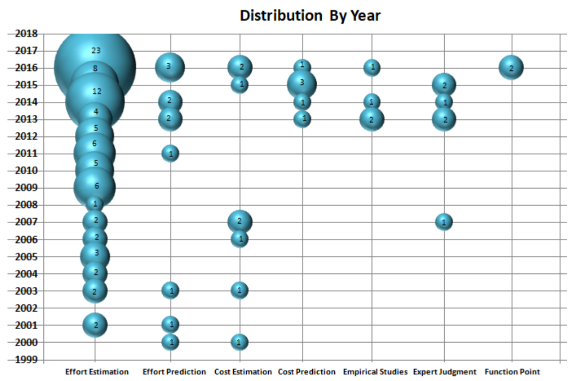

Fig. 6: Distribution of the primary studies based the explored research topics over the years.

6 Threats to validity

Several factors may become a threat to validity of our results. For example, the difficulty to establish a precise relationship between the different kinds of research techniques (while some are based on empirical studies, the others are on statistical studies), identify the feature scope (i.e., detailed or generic) to assure the completeness and precision of the public domain databases and search engines among other aspects [47]. In this sense, we analyzed some threats related to internal, construct and conclusion validity.

Internal validity. It was identified two major threats. First, it was the difficulty in establishing a relationship between the surveyed techniques due to the various existing concepts (e.g., empirical and statistics). This thread was characterized as heterogeneous aspects, which exist between the evaluated techniques. We realized a careful analysis to identify the common features. Next, it was the difficulty to identify the scope of each primary study. To mitigate these threats, we looked for understanding each technique, and then classifying them. The filtering process was performed three times to avoid any bias. All primary studies are listed in Appendix A. In an attempt to ensure that the process of selection of primary studies was as unbiased as possible, it was organized the selection of primary studies as a multi-phase activity, documenting the reasons for the inclusion or exclusion of these and which were the selection criteria, as previously described in Section 2.

Constructor validity. Incorrect classification and deletion of relevant articles are two imminent threats in reviewing the literature. We try to minimize this problem by establishing a review protocol, with well-defined and auditable inclusion and exclusion criteria. Moreover, we have observed other articles not to adopt the good practices highlighted.

Conclusion validity. This threat is strictly related to problems that can affect the reliability of our conclusions. We note that the article selection process may have been influenced by our interests with the study results, as well as our experience in estimating effort in realistic projects. To overcome this problem, inclusion and exclusion criteria were defined to minimize the risk of bias in the selection and filtering process. In addition, the authors have always been aware of possible personal influences. Therefore, this bias was always monitored to reduce negative impacts on the results. Another threat is related to the classification of the works. When conflicting or dubious classifications were found, the authors conducted a collaborative evaluation to reach consensus. In addition, the formulation of search strings is also a process in which some threats can occur, since it is not possible to completely avoid that some specific term has not been considered, simply because the authors do not know any relevant synonym. Finally, all conclusions in this article were made after collecting the results avoiding any possibility of the error rate [47].

7 Related work

The current studies put a huge attention on running empirical studies to produce evidence-based knowledge, and using purely statistical methods to predict effort. However, little has been done to create a systematic map of the published studies in the last decade.

Table 12 Main research venue where the primary studies have been published.

Venue Description	List of primary studies
Journal of Systems and Software	[P04], [P05], [P06], [P13], [P20], [P28], [P31], [P37], [P42], [P58], [P67], [P71], [P82], [P83], [P89], [P116]
Information and Software Technology	[P09], [P10], [P12], [P69], [P80], [P81], [P90], [P119]
IEEE Software	[P01], [P08], [P45], [P51], [P110]
IEEE Trans. on Software Engineering	[P03], [P23], [P29], [P53], [P111]
Int. Conf. on Global Software Engineering	[P16], [P17], [P40], [P49]
Int. Symp. on Emp. Software Engineering and Measurement	[P21], [P32], [P47], [P55]
Int. Conf. on Predictive Models and Data Analytics in SE	[P78], [P79]
IWSM-MENSURA	[P44], [P57], [P02], [P112], [P115]
ACM Trans. on Software Engineering and Methodology	[P70], [P77]
ACM International Conference	[P94], [P97]
Int. Conf. on Evaluation and Assessment in SE	[P73], [P74], [P76]
Int. Conf. on Software Engineering	[P60], [P64]
Int. Symp. on Empirical Software Engineering	[P30], [P63], [P86]
Journal of Software	[P48]
Int. Conf. on Software Process and Product Measurement	[P07]
ACM Symposium on Applied Computing	[P75]
ACM on Hypertext and hypermedia	[P59]
Applied Soft Computing	[P68]
Computational Engineering in Systems Applications	[P46]
Decision Support Systems	[P14]
International Conference on Machine Vision	[P50]
Frontiers in Education Conference	[P35]
Int. Conference on Control and Automation	[P33]
IET Software	[P38], [P88], [P95], [P99]
Int. Journal of Soft. Eng. and Knowledge Engineering	[P52]
Innovations in Soft. Eng. Conference	[P62]
Int. Conf. on Information Syst. and Design of Communication	[P26]
India Software Engineering Conference	[P61]
Int. Conference on Quality Software	[P41]
Research, Innovation, and Vision for the Future	[P24]
Int. Conference on Software and System Processes	[P66]
Int. Software Metrics Symposium	[P39]
Journal of Computing and Information Technology	[P25]
Symposium on Information and Communication Technology	[P65]
Software, Telecommunications and Computer Networks	[P22]
Australian Software Eng. Conf.	[P36]
XP Workshops 2014	[P87]
ICGSE	[P85], [P11], [P18]
ENASE 2016	[P27], [P72]
ENASE 2015	[P56]
ICEIS	[P34]
ICSE	[P92]
ISCBI	[P100]
IJCTA	[P102]
ICGSE 2016	[P93]
SWQD 2016	[P84], [P101]
SNPD 2015	[P54]
Int. Journal of Intell. Systems 31	[P43]
Journal of Web Engineering 8	[P96]
SEAA 2016	[P19]
Artificial Intell. Research 4	[P15]
Procedia Computer Science 89	[P91]
Proceedings - Int. Comp. Soft. and Applications Conference 2	[P98]
ICRITO 2016	[P103]
ICWE'06	[P104]
WMSCI 2006	[P105]
ICCS	[P106]
PROMISE'14	[P107], [P118]
IEE Proceedings - Software	[P108], [P120]
Journal on Information and Management	[P109]
SEW	[P113]
ASWEC	[P114]
EASE'15	[P117]

Jorgensen and Halkjelsvik [P13] carried out empirical studies to understand how change requests of development tasks can impact on the effort estimation to complete them. The authors defined some change requests and then encouraged the participants to implement them. The results highlight that (1) such requests should be discouraged, and (2) the pessimistic estimation should be prioritized. The authors performed four studies, which followed well-detailed guidelines. The result indicated that the use of tool support reduced by about 30% to 40% compared with traditional methods. This study did not evaluate, for example, whether the experiment was performed within a desired time, the goal was to check the variance between the two techniques. These results were confirmed in another study available in [48].

Koch and Mitlohner [59] showed that the iterative and incremental models produced better results compared to other models, such as KLOC (Kilo Lines Of Code), Functions Points Analysis, and COCOMO II (Constructive Cost Model). They claim that this type of iteration has been improperly applied to control incremental development. The research presents an analytic and quantitative framework that assessing the application of incremental approaches and details the impacts on the effort of software development. In the empirical study, the authors developed several techniques to predict the amount of interactions considered ideal for effort estimation in software-development projects. For this reason, these scenarios were detailed and served as a model for the new experiments, and many features were tested as context variables, such as deliveries versus team size, estimate of more elaborated models for further analysis, since the objective was to increase the scope of research and evaluate the impact of agile methodologies over other methodologies. The idea of the authors was to further explore the incremental development techniques and the iteration with other techniques widely used.

Brito et al. [P16] developed an additional study about effort estimation in the context of agile methodologies in a global scale. The estimated effort is a project management activity that is required for running software projects. Despite its importance, there have been few published studies (e.g., [P71]) on such activities within the overall context of agile software development. Moreover, the global development challenges (e.g., different cultures and time zones) were considered important factors. It was found that many challenges that affect the accuracy of estimates of efforts were reported by the participants, and among the chief aspects might be cited: problems regarding the definition of software requirements and technical communication among those involved, and in many cases can harm or help in the project success. In this research, it was detected a lack of categorization of information, just the collected data were saved and evaluated in general terms, rather than considering the detailed context. The results did not present statistic qualifiers, and were not compared to other results studied, since this topic research has been developed by other researchers previously.

Peixoto et al. [P17] demonstrated that in the global software-development context, the effort estimation played a very important role for reducing costs. The authors highlight that it is possible to construct models that analyze existing resources in global software-development projects.

Rosa et al. [P20] replicated an experiment to estimate effort, featuring the project on development attributes and performing compared to a number of similar projects. The evaluation was performed by comparing of several case studies with particular emphasis on the parameterization of properties. The experiment was applied to 59 ERP (Enterprise Resource Planning) systems, and the obtained estimated efforts were compared to the expected results always showing the best results, with an average estimate error of 24% less compared to a real effort. It was noted in this study the absence of analysis and stratification of the collected data. If the stratification was made beforehand, the final results would be more qualified and detailed, providing many possibilities to overcome these and result in a richer and broader context. The area of cost effort estimation in software development has a significant amount of techniques and concepts already tested and validated. These techniques are commonly applied in the productive environment in the software industry. The academic communities in recent years have researched and improved several other techniques, which are being evaluated in a production environment. After reading the articles selected were highlighted some gaps described as follows.

8 Conclusions and future work

This article presented a systematic mapping study about effort estimation. We established a careful research protocol by precisely defining the objective, research questions, search strategy, exclusion and inclusion criteria, data extraction and quality assessment. In total, 120 studies were filtered from a list of 3,746 studies, which were initially retrieved from four widely known electronic databases. The collected results indicate that: (i) multiple effort estimation approaches are more frequently used than just one; (ii) the experimental procedures adopted are not carefully classified or detailed;

IET Research Journals, pp. 1–14

(iii) over 90% of the studies had students the most common participants in their empirical evaluation; (iv) most studies analyzed had their quality assessed as high; and (v) most studies were published in journal.

Some recommendations for future research would be to: increase the breadth of the analysis of the primary studies (Appendix A); refine research about basic software effort estimation issues; conduct a systematic literature review to scrutinize best practices related to specific estimation approaches, technologies, or tools by bringing together information from comparative analysis; run new review studies considering our initial sample of primary studies to ensure that results are based on best-quality evidence; identify a set of relevant properties, which can be used to catalog and classify public databases; refine the list of primary studies by searching manually for more candidate studies, since completeness matters; elaborate new estimation approaches based on fine-grained project properties and supported by machine learning techniques; and carry out more empirical studies to evaluate the estimation approaches, based on expert estimation (e.g., Planning Poker and Delphi), formal estimation model (e.g., COCOMO), multi-objective software effort estimation (e.g., [18]), in real-world settings. In addition, we understand that the use of robust baselines for effort estimation is a practice to be explored in future research, as recently recommended by Sarro and Petrozziello [63]. Still, empirical studies need to be replicated under different technical and cultural settings so that results may be evaluated in terms of their generalization.

Finally, we hope that the results discussed throughout this study can encourage researchers and practitioners to close the gaps described. In addition, this work can be the first step for a more ambitious agenda on how to advance the current literature on effort estimation.

9 Acknowledgment

We thank CNPq grant 313285/2018-7 for partially funding this research.

10 References

1 Chatzipetrou, P., Papatheocharous, E., Angelis, et al.: 'A Multivariate Statistical Framework for the Analysis of Software Effort Phase Distribution', Information and Software Technology, 2015, 59, pp. 149-169
2 Berlin, S., Raz, T., Glezer, C., et al.: 'Comparison of Estimation Methods of Cost and Duration in IT Projects', Information and Software Technology, 2009, **51**, (9), pp. 738-748
3 Seo, Y-S., Yoon, K-A., Bae, D-H.: 'An Empirical Analysis of Software Effort Estimation with Outlier Elimination', Proc. 4th International Workshop on Predictor Models in Software Engineering, May 2008, pp. 25-32
4 Jorgensen, M., Carelius, G. J.: 'An Empirical Study of Software Project Bidding', IEEE Transactions on Software Engineering, 2004, **30**, (12), pp. 953-969
5 'Planning Poker', https://www.planningpoker.com, accessed September 2019
6 'COCOMO', http://csse.usc.edu/csse/research/COCOMOII, accessed September 2019
7 Brown, B. B: 'Delphi Process: A Methodology Used for the Elicitation of Opinions of Experts', RAND Corp, Santa Monica, CA, USA, 1968
8 Menzies, T., Chen, Z., Hihn, J. et al.: 'Selecting Best Practices for Effort Estimation', IEEE Transactions on Software Engineering, 2006, **11**, (32), pp. 883-895
9 Jorgensen, M., Shepperd, M.: 'A Systematic Review of Software Development Cost Estimation Studies', IEEE Trans. on Soft. Engineering, 2007, **33**, (1), pp. 33-53
10 Rodríguez, P., Haghighatkhah, A., Lwakatare, L. E., et al.: 'Continuous Deployment of Software Intensive Products and Services: A Systematic Mapping Study', Journal of Systems and Software, 2017, 123, pp. 263–291
11 Montalvillo, L., Díaz, O.: 'Requirement-Driven Evolution in Software Product Lines: A Systematic Mapping Study', Journal of Systems and Software, 2016, 122, pp. 110-143
12 Ahmad, A., Babar, M. A.: 'Software Architectures for Robotic Systems: A Systematic Mapping Study', Journal of Systems and Software, 2016, 122, pp. 16-39
13 Qiu, D., Li, B., Ji, S., Leung, H.: 'Regression Testing of Web Service: A Systematic Mapping Study', ACM Computing Surveys, 2014, **47**, (2), pp. 1-46
14 Al-Qudah, S., Meridji, K., Al-Sarayreh, K. T.: 'A Comprehensive Survey of Software Development Cost Estimation Studies', International Conference on Intelligent Information Processing, Security and Advanced Communication, November 2015, pp. 1-5
15 Keung, J.: 'Empirical Evaluation of Analogy-X for Software Cost Estimation', Proc. International Symposium on Empirical Software Engineering and Measurement, Kaiserslautern, Germany, 2008, pp. 294-296
16 Jalali, O., Menzies, T., Baker, D., et al.: 'Column Pruning Beats Stratification in Effort Estimation', Proc. 3rd International Workshop on Predictor Models in Software Engineering, Minneapolis, MN, USA, 2007, pp. 1-9
17 Kitchenham, B. A., Charters, S.: 'Guidelines for Performing Systematic Literature Reviews in Software Engineering', (Keele University, 2007), pp. 1-65
18 Sarro, F., Petrozzielloy, A., Harman, M.: 'Multi-objective Software Effort Estimation', Proc. 38th International Conference on Software Engineering, Austin, Texas, USA, May 2016, pp. 619-630
19 Petersen, K., Vakkalanka, S., Kuzniarz, L.: 'Guidelines for Conducting Systematic Mapping Studies in Software Engineering: An Update', Information and Software Technology, 2015, 64, pp. 1-18
20 Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M.: 'Systematic Mapping Studies in Software Engineering', Proc. 12th International Conference on Evaluation and Assessment in Software Engineering, Bari, Italy, June 2008, pp. 68-77
21 Fernández-Sáez, A. M., Genero, M., Chaudron, M.: 'Empirical Studies Concerning the Maintenance of UML Diagrams and their Use in the Maintenance of Code: A Systematic Mapping Study', Information and Software Technology, 2013, **55**, (7), pp. 1119-1142
22 Cooper, D.: 'What is a mapping study?', Journal of the Medical Library Association, 2016, 104, pp. 76–78
23 Kitchenham, B., Mendes, E.: 'Why Comparative Effort Prediction Studies may be Invalid', Proc. 5th International Conference on Predictor Models in Software Engineering, British Columbia, Canada, May 2009, pp. 1-5
24 Nguyen, V.: 'Improved Size and Effort Estimation Models for Software Maintenance', Information and Software Technology, 2008, 50, pp. 833-859
25 Kocaguneli, E., Menzies, T., Keung, J., et al.: 'Active Learning and Effort Estimation: Finding the Essential Content of Software Effort Estimation Data', IEEE Transactions on Software Engineering, 2013, **39**, (8), pp. 1040-1053
26 Dyba, T., Dingsoyr, T.: 'Empirical Studies of Agile Software Development: A Systematic Review', Information and Software Technology, 2008, 50, pp. 833-859
27 Kocaguneli, E., Menzies, T., Bener, A. et al.: 'Exploiting the Essential Assumptions of Analogy-based Effort Estimation', IEEE Transactions on Software Engineering, 2011, **38**, (2), pp. 425-438
28 Kocaguneli, E., Menzies, T.: 'How to Find Relevant Data for Effort Estimation?', LCSEE, West Virginia Univ, Morgantown, 2011, 38, pp. 255-264
29 Dyba, T., Dingsoyr, T.: 'Empirical Studies of Agile Software Development: A Systematic Review', Information and software technology, **50**, (9-10), pp. 833-859
30 Mendes, T.S., de F Farias, M.A., Mendonça, M., et al.: 'Impacts of Agile Requirements Documentation Debt on Software Projects: A Retrospective Study', 31st Annual ACM Symposium on Applied Computing, Gyeongju, Korea, March 2014, pp. 1290-1295
31 Jorgensen, M., Grimstad, S.: 'Over-optimism in Software Development Projects: The winner's curse', Information and Software Technology, 2008, 50, pp. 833-859
32 Mendes, E., Turhan, B., Rodríguez, P., et al.: 'Estimating the Value of Decisions Relating to Managing and Developing Software-intensive Products and Projects', 11th International Conference on Predictive Models and Data Analytics in Software Engineering, 2015, pp. 1-7
33 Huijgens, H., Vogelezang, F.: 'Do Estimators Learn? On the Effect of a Positively Skewed Distribution of Effort Data on Software Portfolio Productivity', 7th International Workshop on Emerging Trends in Software Metrics, 2016, pp. 8-14
34 Gupta, S., Sikka, G., Verma, H.: 'Recent Methods for Software Effort Estimation by Analogy', ACM SIGSOFT Software Engineering Notes, 2011, 4, pp. 1-5
35 Jeffery, R., Ruhe, M., Wieczorek, I.: 'Using Public Domain Metrics to Estimate Software Development Effort', Proc. 7th International Software Metrics Symposium, London, England, UK, April 2001, pp. 16-27
36 Briand, L.C., Wüst, J.: 'Modeling Development Effort in Object-Oriented Systems Using Design Properties', IEEE Transactions on Software Engineering, 2011, **11**, (27), pp. 963-986
37 Song, L., Minku, L.L., Yao, X.: 'The Potential Benefit of Relevance Vector Machine to Software Effort Estimation', Proc. 10th International Conference on Predictive Models in Software Engineering, Torino, Italy, September 2014, pp. 1-10
38 Saif, A., Abbas, S., Fayed, Z.: 'Finding The Best Software Project Options by PDBO Algorithm for Improving Software Development Effort, Time and Quality', Proc. 10th International Conference on Informatics and Systems, Giza, Egypt, May 2016, pp. 304-311
39 Ramacharan, S., Rao, K.V.G.: 'Software Effort Estimation of GSD Projects Using Calibrated Parametric Estimation Models', Proc. 2nd International Conference on Information and Communication Technology for Competitive Strategies, Udaipur, India, March 2016, pp. 1-8
40 Sigweni, B., Shepperd, M., Turchi, T.: 'Realistic Assessment of Software Effort Estimation Models', Proc. 20th International Conference on Evaluation and Assessment in Software Engineering, Limerick, Ireland, June 2016, pp. 1-6
41 Mittas, N., Angelis, L.: 'Ranking and Clustering Software Cost Estimation Models through a Multiple Comparisons Algorithm', IEEE Transactions on Software Engineering, 2012, 39, pp. 537-551
42 Hamdan, K., Bibi, S., Angelis, L. and Stamelos, I.: 'A Bayesian Belief Network Cost Estimation Model that Incorporates Cultural and Project leadership Factors', Proc. IEEE Symposium on Industrial Electronics and Applications, Kuala Lumpur, Malaysia, October 2009, pp. 985-989
43 Azzeh, M., Neagu, D. and Cowling, P.: 'Improving Analogy Software Effort Estimation using Fuzzy Feature Subset Selection Algorithm', Proc. 4th International Workshop on Predictor Models in Software Engineering, Leipzig, Germany, May 2008, pp. 71-78
44 Kumar, G. and Bhatia, P. K.: 'Neuro-Fuzzy Model to Estimate and Optimize Quality and Performance of Component Based Software Engineering', ACM SIGSOFT

Software Engineering Notes, 2015, **40**, (2), pp. 1-6
45 Briand L.C., Langley T., Wieczorek, I.: 'A replicated Assessment and Comparison of Common Software Cost Modeling Techniques', Proc. 22nd International Conference on Software Engineering, San Francisco CA, USA, May 2013, pp. 377-386
46 Salman I, Misirli AT, Juristo N.: 'Are Students Representatives of Professionals in Software Engineering Experiments?', Proc. 37th IEEE International Conference on Software Engineering, Florence Italy, May 2015, pp. 666-676
47 Wohlin, C., Runeson, P., Host, M., et al.: 'Experimentation in Software Engineering'. Springer Science and Business Media, 2012
48 Stammel, J., Trifu, M.: 'Tool-Supported Estimation of Software Evolution Effort in Service-Oriented Systems', Proc. 5th International Workshop on Model-Driven Software Migration (MDSM), and 5th International Workshop on System Quality and Maintainability, Copenhagen, Denmark, October 2011, pp. 10-14
49 Basili, V., Caldiera, G., Rombach, H.: 'The Goal Question Metric Paradigm', Encyclopedia of Software Engineering, 1994, Wiley, New York
50 Gonçales, L.J., Farias, K., Scholl, M., et al.: 'Comparison of Design Models: A Systematic Mapping Study', International Journal on Software Engineering and Knowledge Engineering, 2015, 25, pp. 1765-1769
51 Lau, J., Ioannidis, J.P., Schmid, C.H.: 'Quantitative Synthesis in Systematic Reviews', 1997, Annals of Internal Medicine, **127**, (9), pp. 820-826
52 Muller, S. C.: 'Measuring software developers' perceived difficulty with biometric sensors', 37th IEEE International Conference on Software Engineering, Florence, Italy, May 2015, pp. 887-890
53 Fritz, T., Muller, S. C.: 'Leveraging biometric data to boost software developer productivity', Proc. 23rd International Conference on Software Analysis, Evolution, and Reengineering, Osaka, Japan, March 2016, Vol. 5, pp. 66-77
54 Crk, I., Kluthe, T. and Stefik, A.: 'Understanding programming expertise: an empirical study of phasic brain wave changes', ACM Transactions on Computer-Human Interaction, 2016, **23**, (1), pp. 2-29
55 Muller, S. C., Fritz, T.: 'Using (bio) metrics to predict code quality online', Proc. 38th International Conference on Software Engineering, Austin, Texas, USA, May 2016, pp. 452-463
56 Muller, S. C., Fritz, T.: 'Stuck and Frustrated or in Flow and Happy: Sensing Developers' Emotions and Progress', Proc. 37th IEEE International Conference on Software Engineering, Florence, Italy, May 2015, Vol. 1, pp. 688-699
57 Wieringa, R., Maiden, N., Mead, N., et al.: 'Requirements Engineering Paper Classification and Evaluation Criteria: A Proposal and a Discussion', Requirements engineering, 2016, **11**, (1), pp. 102–107
58 Bertolino, A., Calabro, A., Lonetti, F. et al.: 'A Categorization Scheme for Software Engineering Conference Papers and its Application,' Journal of Systems and Software, 2018, 137, pp. 114-129
59 Benediktsson, O., and Dalcher, D.: Effort Estimation in Incremental Software Development, IEE Proceedings-Software, 2003, **150**, (6), pp. 351–357
60 Bischoff, V., Farias, K., Gonçales, L., et al.: 'Integration of feature models: A systematic mapping study,' Information and Software Technology, 2019, 105, pp 209-225
61 Gonçales, L. J., Farias, K., Oliveira, T.C., et al.: 'Comparison of Software Design Models: An Extended Systematic Mapping Study', ACM Computing Surveys (CSUR), 2019, 52,(3), pp. 1–41
62 Gonçales, L.J., Farias, K., Silva, B., et al.: 'Measuring the Cognitive Load of Software Developers: A Systematic Mapping Study', Proc. 27th International Conference on Program Comprehension, Montreal, Canada, May 2019, pp. 42–52
63 Sarro, F., Petrozziello, A.: 'Linear programming as a baseline for software effort estimation', ACM Transactions on Software Engineering and Methodology (TOSEM), 2018, **1**, (2), pp. 1-7

11 Appendix A - List of primary studies

The articles selected as primary studies in the systematic mapping study are listed below.

[P01] M. Jorgensen, Practical Guidelines for Expert-Judgment-Based Software Effort Estimation, IEEE Software, 22(3):57–62, 2005.
[P02] M. Usman, R. Britto: Effort Estimation in Co-located and Globally Distributed Agile Software Development: A Comparative Study, IWSM-Mensura, pp. 219–224, 2016.
[P03] S. Grimstad, J. M., Software Development Estimation Biases: The Role of Interdependence, IEEE Trans. on Software Engineering, 38(3):677-693, 2012.
[P04] G. M. Stephen, M. Shepperd, Combining Techniques to Optimize Effort Predictions in Software Project Management, Journal of Systems and Software, 66(2):91-98, 2003.
[P05] S. Grimstad, M. Jorgensen, Inconsistency of Expert Judgment-Based Estimates of Software Development Effort, Journal of System and Software, 80(11):1770–1777, 2007.
[P06] C. Wallshein, A. G. Loerch, Software Cost Estimating for CMMI Level 5 Developers, Journal of System and Software, 105(C):72–78, 2015.
[P07] S. Tunalilar, O. Demirors, EFES: An Effort Estimation Methodology, IWSM, pp. 205–214, 2012.
[P08] M. Jorgensen, Relative Estimation of Software Development Effort: It Matters with What and How You Compare, IEEE Software, 30(2):74–79, 2013.
[P09] M. Jorgensen, The Influence of Selection Bias on Effort Overruns in Software Development Projects, Information and Software Technology, 87(9):1640–1650, 2013.
[P10] M. Jorgensen, D. I. K. Sjoberg, Impact of Effort Estimates on Software Project Work, Elsevier, pp. 939–948, 2001.
[P11] R. Britto, E. Mendes, J. Borstler, An Empirical Investigation on Effort Estimation in Agile Global Software Development, 10th International Conference on Global Software Engineering, pp. 38–45, 2015
[P12] A. Idri, F. A. Amazal, A. Abran, Analogy-based Software Development Effort Estimation: A Systematic Mapping and Review, Information & Software Technology, Vol. 58, pp. 206–230, 2015.
[P13] M. Jorgensen, T. Halkjelsvik, The Effects of Request Formats on Judgment-Based Effort Estimation, Journal of Systems and Software, 83(1):29–36, 2010.
[P14] S. Koch, J. Mitlohner, Software Project Effort Estimation with Voting Rules, Decision Support Systems, 46(4):895–901, 2009.
[P15] F.H. Nayebi, A. Abran, J. Desharnais: Automated Selection of a Software Effort Estimation Model Based on Accuracy and Uncertainty, Artificial Intell. Research, 4(2):45-54, 2015.
[P16] R. Britto, E. Mendes, B. Borstler, An Empirical Investigation on Effort Estimation in Agile Global Software Development, 10th International Conference on Global Software Engineering, pp. 38–45, 2015.
[P17] C. Peixoto, J. Audy, R. Pricladnicki, Effort Estimation in Global Software Development Projects: Preliminary Results from a Survey, 5th IEEE International Conference on Global Software Engineering, pp. 123–127, 2010.
[P18] R. Britto, V. Freitas, E. Mendes, M. Usman: Effort Estimation in Global Software Development: A Systematic Literature Review, 9th International Conference on Global Software Engineering, pp. 135–144, 2014.
[P19] S. E. Koutbi, A. Idri, A. Abran: Systematic Mapping Study of Dealing with Error in Software Development Effort Estimation, 42th Euromicro Conference on Software Engineering and Advanced Applications, pp. 140–147, 2016.
[P20] W. Rosa, T. Packarda, A. Krupananda, J. Bilbrob, M. Hodal, COTS Integration and Estimation for ERP, Journal of Systems and Software, 86(2):538-550, 2013.
[P21] D. Basten, W. Mellis, A Current Assessment of Software Development Effort Estimation, Int. Symp. on Empirical Software Engineering and Measurement, pp. 235–244, 2011.
[P22] H. Karna, S. Gotovac, Modeling Expert Effort Estimation of Software Projects, SoftCOM, pp. 356–360, 2014.
[P23] M. Jorgensen et al., Reasons for Software Effort Estimation Error: Impact of Respondent Role, Information Collection Approach, and Data Analysis Method, IEEE Transactions on Software Engineering, 30(12):993–1007, 2004.
[P24] M. Yahya et al., Effects of Software Process Maturity on COCOMO II Effort Estimation from CMMI Perspective, IEEE International Conference on Research, Innovation and Vision for the Future in Computing and Communication Technologies, pp. 255–262, 2008.
[P25] I. Alsmadi, M. S. Nuser, Evaluation of Cost Estimation Metrics: Towards a Unified Terminology, Journal of Computing and Information Technology, 21(1):23–24, 2013.
[P26] D. Kashyap, A. K. Misra, Software Development Cost Estimation Using Similarity Difference between Software Attributes, International Conference on Information Systems and Design of Communication, pp. 729–741, 2013.
[P27] A. Idri, M. Hosni, A. Abran, Systematic Mapping Study of Ensemble Effort Estimation, ENASE 2016: 132–139.
[P28] A. Idri, M. Hosni, A. Abran, Systematic Literature Review of Ensemble Effort Estimation. Journal of Systems and Software, Vol. 118, pp. 151-175, 2016.
[P29] M. Jorgensen, M. Shepperd, A Systematic Review of Software Development Cost Estimation Studies, IEEE Trans. on Software Engineering, 33(1):33-53, 2007.
[P30] K. Molokken-Ostvold, M. Jorgensen, A Review of Surveys on Software Effort Estimation, ESEM, pp. 223–230, 2003.
[P31] M. Jorgensen, A Review of Studies on Expert Estimation of Software Development Effort, Journal of System and Software, pp. 37–60, 2004.
[P32] S. Macdonell, M. Shepperd, Comparing Local and Global Software Effort Estimation Models: Reflections on a Systematic Review, 1st International Symposium on Empirical Software Engineering and Measurement, pp. 401–409, 2007.
[P33] W. Han, T. Lu, H. Jiang, W. Li, A Study on the Improvement of Software Effort Estimation, IEEE ICCA, IEEE Computer Society, 2014.
[P34] D.T. M. C. Branco, E. C. C .Oliveira, L. Galvao, R. Prikladnicki, T. Conte: An Empirical Study about the Influence of Project Manager Personality in Software Project Effort, ICEIS (2), pp. 102–113, 2015
[P35] L. Alves, A. Souza, P. Ribeiro, R. Machado, An Empirical Study on the Estimation of Software Development Effort with Use Case Points, Frontiers in Education Conference, pp. 101–107, 2013.
[P36] J. Verner, B. Henderson-Sellers, A Pilot study in effort estimation for the generalization of object-oriented components for reuse, Software Engineering Conference, pp. 190–199, 2001.
[P37] A. Idri, I. Abnane, A. Abran: Missing Data Techniques in Analogy-based Software Development Effort Estimation, Journal of Systems and Software, Vol. 117, pp. 595–611, 2016.
[P38] T. Wijiayasiriwardhane, R. Lai, K. Kang, Effort Estimation of Component-Based Software Development: A Survey, IET Software, 5(2):216–228, 2010.
[P39] S. Grimstad, M. Jorgensen, K. Molokken-Ostvold, The Clients´ Impact on Effort Estimation Accuracy in Software Development Projects, METRICS'05, pp.1–10, 2005.
[P40] M. Bajta, Analogy-Based Software Development Effort Estimation in Global Software Development, International Conference on Global Software Development, pp. 51–54, 2015.
[P41] K. Furulund, K. Molokken-Ostvold, Increasing Software Effort Estimation Accuracy: Using Experience Data, Estimation Models and Checklists, QSIC, pp. 1-6, 2007.
[P42] B. Nauman, K. Petersen, C. Wohlin, A Systematic Literature Review on the Industrial Use of Software Process Simulation, Journal of Systems and Software, 97:65-85, 2014.
[P43] A. Idri, F. A. Amazal, A. Abran, Accuracy Comparison of Analogy-Based Software Development Effort Estimation Techniques. International Journal Intelligent System, 31(2):128-152, 2016.
[P44] O. Top, B. Ozkan, M. Nabi, O. Demirors, Internal and External Software Benchmark Repository Utilization for Effort Estimation, IWSM and MESURA, pp.302–307, 2011.
[P45] M. Jorgensen, B. Boehm, S. Rifkin, Software Development Effort Estimation: Formal Models or Expert Judgment?, IEEE Software, 26(2):15–19, 2009.
[P46] A. Ferchichi, J. Bourey, M. Bigand, M. Barron, Design Systems Engineering of Software Products: Implementation of a Software Estimation Model, CESA'06, pp. 1181-–1188, 2006.
[P47] A. Girasella, F. Pagin, An UML-based Approach to Software Development Cost Estimation, ACM/IEEE Int. Symp. on Empirical Software Engineering and Measurement, pp. 1—5, 2014.
[P48] W. Han, T. Lu, X. Zhang, A New Estimation Model for Small Organic Software Project, Journal of Software, pp. 2218-–2222, 2013.
[P49] A. Lamersdorf, J. Munch, A. F. V. Torre, C. R. Sanchez, D. Rombach, Estimating the Effort Overhead in Global Software Development, International Conference on Global Software Development, pp. 267—276, 2010.
[P50] V. Khatibi, E. Khatibi, Investigating the Effect of Software Project Type on Accuracy of Software Development Effort Estimation in COCOMO Model, ICMV'11, pp. 1-7, 2012.
[P51] M. Jorgensen, What We Do and Don't Know about Software Development Effort Estimation, IEEE Software, 31(2):37–40, 2014.
[P52] C. Arumugam, C. Babu, Developmental Size Estimation For Object-Oriented Software Based on Analysis Model, Int. Journal of Soft. Eng. and Knowledge Engineering, 23(3):289–308, 2013.
[P53] F. Fioravanti, P. Nesi, Estimation and Prediction Metrics for Adaptive Maintenance Effort of Object Oriented Systems, IEEE Trans. on Software Engineering, 27(2):1062–1084, 2001.
[P54] A. Idri, I. Abnane, A. Abran: Systematic Mapping Study of Missing Values Techniques in Software Engineering Data, SNPD 2015: 671–678.
[P55] P. Abrahamsson, I. Fronza, R. Moser, W. Vlasenko, J. and Pedrycz, Predicting Development Effort from User Stories, International Symposium on Empirical Software Engineering and Measurement, pp. 400–403, 2011.
[P56] M. E. Bajta, A. Idri, J. L. F. Alemán, J.N. Ros, A. Toval, Software Cost Estimation for Global Software Development - A Systematic Map and Review Study, ENASE, pp. 197–206, 2015.
[P57] F. A. Amazal, A. Idri, A. Abran, An Analogy-based Approach to Estimation of Soft. Development Effort Using Categorical Data, IWSM/Mensura, pp. 252-261, 2014.
[P58] A. Idri, M. Hosni, A. Abran, Systematic literature review of ensemble effort estimation,The Journal of Systems and Software, Vol. 118, pp. 151–175, 2016.

[P59] E. Mendes, W. Hall, Towards the Prediction of Development Effort for Web Applications, pp. 242–243, 2000.
[P60] L. L. Minku, X. Yao, How to Make Best Use of Cross-Company Data in Software Effort Estimation?, ICSE, pp. 446–456, 2014.
[P61] S. M. Satapathy, S. K. Rath, Class Point Approach for Software Effort Estimation Using Various Support Vector Regression Kernel Methods, ISEC, pp. 1–10, 2014.
[P62] S. M. Satapathy, S. K. Rath, Empirical Assessment of Machine Learning Models for Effort Estimation of Web-based Applications, ISEC, pp. 74–84, 2016.
[P63] E. Oliveira, T. Conte, M. Cristo, E. Mendes, Software Project Managers' Perceptions of Productivity Factors: Findings from a Qualitative Study, pp. 1–6, 2016.
[P64] F. Sarro, A. Petrozziello, M. Harman, Multi-Objective Software Effort Estimation, Proceedings of the 38th International Conference on Software Engineering, pp. 619–630, 2016.
[P65] S. Basri, N. Kama, F. Haneem, S. A. Ismail,Predicting Effort for Requirement Changes during Software Development, SoICT, pp. 380-387, 2016.
[P66] B. Tanveer, L. Guzmán, U. M. Engel, Understanding and Improving Effort Estimation in Agile Software Development - An Industrial Case Study, IEEE/ACM ICSSP, pp. 41-49, 2016.
[P67] A. Idri, I. Abnane, A. Abran, Missing Data Techniques in Analogy-based Software Development Effort Estimation, Journal of Systems and Software, Vol. 117, pp. 595–611, 2016.
[P68] A. Idri, M. Hosnia, A. Abran, Improved Estimation of Software Development Effort Using Classicaland Fuzzy Analogy Ensembles, Applied Soft Computing, 49, pp. 990-1019, 2016.
[P69] A. Idri, F. A. Amazal, A. Abran, Analogy-based Software Development Effort Estimation: A Systematic Mapping and Review, Information and Software Technology, Vol. 58, pp. 206–230, 2015.
[P70] P. A. Whigham, C. A. Owen, S. G. Macdonell, A Baseline Model for Software Effort Estimation, ACM Trans. on Software Eng. and Methodology, Vol. 24, pp. 1-11, 2015.
[P71] N. Mittas, E. Papatheocharous, L. Angelis, A.S. Andreou, Integrating Non-parametric Models with Linear Components for Producing Software Cost Estimations, Journal of Systems and Software, Vol. 99, pp. 120–134, 2015.
[P72] A. Idri, M. Hosni, A. Abran, Systematic Mapping Study of Ensemble Effort Estimation, ENASE pp. 132–139, 2016.
[P73] E. Mendes, M. Kalinowski, D. Martins, F. Ferrucci, F. Sarro, Cross- vs. Within-Company Cost Estimation Studies Revisited: An Extended Systematic Review, EASE, pp. 1–10, 2014.
[P74] O. Matos, T. Conte, E. Mendes, Is there a Place for Qualitative Studies when Identifying Effort Predictors? A Case in Web Effort Estimation, EASE'14, pp. 1–10, 2014.
[P75] M. P. Basgalupp, R. C. Barros, T. S. da Silva, Software Effort Prediction: A Hyper-Heuristic Decision-Tree based Approach, SAC'13, pp. 18–22, 2013.
[P76] O. Matos, L. Fortaleza, T. Conte, E. Mendes, Realising Web Effort Estimation: A Qualitative Investigation, EASE'13, pp. 12–23, 2013.
[P77] L. L. Minku, X. Yao, Software Effort Estimation as a Multiobjective Learning Problem, ACM Trans. on Software Engineering and Methodology, Vol. 22, pp. 1-32, 2013.
[P78] F. Ferrucci, E. Mendes, F. Sarro, Web Effort Estimation: the Value of Cross-company Data Set Compared to Single-company Data Set, PROMISE, 2012.
[P79] M. V. Kosti, N. Mittas, L. Angelis, Alternative Methods Using Similarities in Software Effort Estimation, PROMISE, pp. 59-68, 2012.
[P80] K. Petersen, Measuring and Predicting Software Productivity: A Systematic Map and Review, Information and Software Technology, Vol. 53, pp. 317–343, 2011.
[P81] Z. Muzaffar, M. A. Ahmed, Software Development Effort Prediction: A Study on the Factors Impacting the Accuracy of Fuzzy Logic Systems, Information and Software Technology, Vol. 52, pp. 92–109, 2010.
[P82] M. Jorgensen: Unit Effects in Software Project Effort Estimation: Work-hours Gives Lower Effort Estimates than Workdays, Journal of Systems and Software, Vol. 117, pp. 274–281, 2016.
[P83] E. Lohre, M. Jorgensen, Numerical Anchors and their Strong Effects on Software Development Effort Estimates, Journal of Systems and Software, Vol. 116, pp. 49–56, 2016.
[P84] M. Jorgensen: The Use of Precision of Software Development Effort Estimates to Communicate Uncertainty. SWQD 2016: 156–168.
[P85] R. Britto, E. Mendes, C. Wohlin: A Specialized Global Software Engineering Taxonomy for Effort Estimation. ICGSE 2016: 154–163.
[P86] E. Kocaguneli, T. Menzies, E. Mendes: Transfer Learning in Effort Estimation, Empirical Software Engineering, 20(3):813–843, 2015.
[P87] R. Britto, M. Usman, E. Mendes: Effort Estimation in Agile Global Software Development Context, XP Workshops, pp. 182–192, 2014.
[P88] S. M. Satapathy, P. A.Barada, K.R. Santanu, Early Stage Software Effort Estimation Using Random Forest Technique Based on Use Case Points, IET Software, 10(1):10–17, 2016.
[P89] F. L. Gonzalez, M.F. Diego, C. Loka, The Usage of ISBSG Data Fields in Software Effort Estimation: A Systematic Mapping Study, Journal of Systems and Software, Vol. 113, pp. 188-215, 2016.
[P90] C. Lokan, E. Mendes, Investigating the Use of Duration-based Moving Windows to Improve Software Effort Prediction: A Replicated Study, Information and Software Technology, Vol. 56, Issue 9, pp. 1063–1075, September 2014.
[P91] P. Rijwani, J. Sonal, Enhanced Software Effort Estimation Using Multi-layered Feed Forward Artificial Neural Network Technique, Procedia Computer Science, Vol. 89, pp. 307-312, 2016.
[P92] P. Mohagheghi, B. Anda, R. Conradi, Effort Estimation of Use Cases for Incremental Large–scale Software Development, In: 27th International Conference on Software Engineering, pp. 303-311, 2005.
[P93] R. Britto, E. Mendes, C. Wohlin, A Specialized Global Software Engineering Taxonomy for Effort Estimation, 11th IEEE International Conference on Global Software Engineering, pp. 154-163, 2016.
[P94] M. Usman, E. Mendes, J. Borstler, Effort Estimation in Agile Software Development: A survey on the State of the Practice, 2015, ACM International Conference Proceeding Series, pp. 27–29, April 2015.
[P95] C. Lokan, E. Mendes, Investigating the Use of Chronological Split for Software Effort Estimation, IET Software, Vol. 3, pp. 422-434, 2009.
[P96] S.D.I. Martino, F. Ferrucci, C. Gravino, E. Mendes, Measures and Techniques for Effort Estimation of Web Applications: An Empirical Study Based on a Single-company Dataset, Journal of Web Engineering, 8(2):154-181, 2009.
[P97] S. Basri, N. Kama, F. Haneem, S.A. Ismail, Predicting Effort for Requirement Changes during Software Development, International Conference Proceeding Series, pp. 380-387, December 2016.
[P98] K. Moharreri, A.V. Sapre, J. Ramanathan, R. Ramnath, Cost-Effective Supervised Learning Models for Software Effort Estimation in Agile Environments, International Computer Software and Applications Conference 2, pp. 135–140, 2016.
[P99] S. M. Satapathy, B.P. Acharya, S.K. Rath, Early Stage Software Effort Estimation Using Random Forest Technique Based on Use Case Points, IET Software, 10(1):10–17, 2016.
[P100] S. Tariq, M. Usman, R. Wong, Y. Zhuang, S. Fong, On Learning Software Effort Estimation, 3rd International Symposium on Computational and Business Intelligence, pp. 79–84, 2016.
[P101] M.Jorgensen, The Use of Precision of Software Development Effort Estimates to Communicate Uncertainty, International Conference on Software Quality, pp 156-168, 2016.
[P102] D. Nandal, O. P. A. Sangwan, Survey Report on Various Software Estimation Techniques and Practices, International Journal of Control Theory and Applications, 9(22):75–83, 2016.
[P103] S. K. Khatri, S. Malhotra, P. Johri, Use Case Point Estimation Technique in Software Development, 5th International Conference on Reliability, Infocom Technologies and Optimization (trends and future directions) (ICRITO), pp. 123-128, 2016.
[P104] G. Costagliola, S. Di Martino, F. Ferrucci, G. Tortora, G. Vitiello, Effort estimation Modeling Techniques: A Case Study for Web Applications, 6th International Conference on Web Engineering pp. 9-16, 2006.
[P105] M. A. Yahya, F. A. M. Masoud, A. Hudaib, The Effects of Software Development Environment on Software Cost Estimation, 10th World Multi-Conference on Systemics, Cybernetics and Informatics, Jointly with the 12th International Conference on Information Systems Analysis and Synthesis, pp. 239-243, 2006.
[P106] Y. Zheng, B. Wang, Y. Zheng, L. Shi, Estimation of Software Projects Effort Based on Function Point, 4th International Conference on Computer Science and Education, pp. 941–943, 2009.
[P107] U. Muhammad, E. Mendes, F. Weidt, R. Britto, Effort Estimation in Agile Software Development: A Systematic Literature Review, PROMISE, pp. 82-91, 2014.
[P108] O. Benediktsson, D. Dalcher, Effort Estimation in Incremental Software Development, IEE Proceedings - Software, 150(6):351–357, 2003.
[P109] I. Stamelos, L. Angelis, M. Morisio, E. Sakellarisc, G. L. Bleris, Estimating the Development Cost of Custom Software, Journal on Information and Management, 40(8): 729-741, 2003.
[P110] C. Ebert, H. Soubra, Functional Size Estimation Technologies for Software Maintenance, IEEE Software, pp. 24–28, 2014.
[P111] B. Boehm, C. Abts, S. Chulani, Software Development Cost Estimation Approaches: A Survey, Software Engineering, pp. 177–205, 2000.
[P112] D. Moulla, I. Damakoa, D. Kolyang, Application of Function Points to Software Based on Open Source: A Case Study, Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement, pp. 191—195, 2014.
[P113] J. Cunha, S. Cruz, M. Costa, A. Rodrigues, Implementing Software Effort Estimation in a Medium-sized Company, IEEE Soft. Engineering Workshop (SEW), pp.92-–96, 2011.
[P114] P. Naunchan, D. Sutivong, Adjustable Cost Estimation Model for COTS-based Development, ASWEC, pp. 1—8, 2007.
[P115] C. Wille, A. Fiegler, R. Neumann, R. Dumke, Evidence-Based Evaluation of Effort Estimation Methods, Software Measurement, Joint Conference of the International Workshop on Software Measurement and the International Conference on Software Process and Product Measurement, pp. 196—205, 2011.
[P116] M. Azzeh, D. Neagu and P. I. Cowling, Analogy-based Software Effort Estimation Using Fuzzy Numbers, Journal of Systems and Software, Vol. 84, pp. 270-284, 2011.
[P117] M. Usman, E. Mendes, J. Borstler, Effort Estimation in Agile Software Development: A Survey on the State of the Practice, 19th International Conference on Evaluation and Assessment in Software Engineering, pp. 1–10, 2015.
[P118] M. Usman, E. Mendes, F. Weidt, R. Britto, Effort Estimation in Agile Software Development: A Systematic Literature Review, PROMISE, pp. 82–91, 2014.
[P119] S. Di Martino, F. Ferrucci, C. Gravino, F. Sarro. Web Effort Estimation: Function point analysis vs. COSMIC, Information and Software Technology, Vol. 72, pp. 90–109, 2016.
[P120] H. Anandi, B. Boehm, Function Point Analysis for Software Maintenance, 10th International Symposium on Empirical Software Engineering and Measurement, pp. 1–6, 2016.

12 Appendix B - Quality assessment

Table 13 Results of the quality assessment

Id	Q1	Q2	Q3	Q6	Q5	Q6	Total	Score
P01	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P02	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P03	3,5	2,0	2,0	4,0	4,0	4,0	19,5	High
P04	3,0	2,0	1,5	3,0	3,5	3,5	16,5	High
P05	3,0	2,0	2,0	4,0	3,5	4,0	18,5	High
P06	3,0	1,5	2,0	2,5	3,0	3,0	15	Medium
P07	3,5	1,5	1,0	2,5	3,0	3,0	14,5	Medium
P08	3,5	2,0	2,0	4,0	4,0	3,5	19	High
P09	3,5	2,0	2,0	4,0	4,0	4,0	19,5	High
P10	3,5	2,0	2,0	4,0	3,5	3,5	18,5	High
P11	3,5	2,0	2,0	3,5	3,5	4,0	18,5	High
P12	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P13	3,0	1,5	2,0	3,0	4,0	3,5	17	High
P14	3,0	1,5	2,0	3,5	3,5	3,5	17	High
P15	3,0	2,0	2,0	2,5	2,5	2,5	14,5	Medium
P16	4,0	1,5	2,0	4,0	4,0	4,0	19,5	High
P17	3,0	2,0	1,5	3,0	3,5	3,5	16,5	High
P18	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P19	3,0	2,0	2,0	2,5	2,5	2,5	14,5	Medium
P20	3,5	2,0	2,0	4,0	3,5	4,0	19	High
P21	2,5	2,0	2,0	2,0	3,0	3,0	14,5	Medium
P22	2,0	1,5	2,0	3,0	2,5	3,5	14,5	Medium
P23	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P24	3,0	2,0	1,5	3,5	4,0	3,5	17,5	High
P25	3,5	2,0	2,0	3,5	3,5	3,5	18	High
P26	2,0	1,0	1,0	2,0	2,0	2,0	10	Low
P27	3,0	2,0	2,0	3,0	2,5	3,0	15,5	Medium
P28	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P29	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P30	3,0	2,0	2,0	3,5	3,0	3,5	17	High
P31	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P32	3,5	2,0	2,0	3,5	3,5	3,5	18	High
P33	2,5	1,0	2,0	3,5	3,0	3,0	15	Medium
P34	4,0	2,0	2,0	3,5	4,0	4,0	19,5	Medium
P35	3,5	2,0	2,0	2,5	2,5	2,5	15	Medium
P36	2,5	1,5	2,0	2,5	3,0	3,0	14,5	Medium
P37	4,0	2,0	2,0	3,5	3,5	3,5	18,5	High
P38	3,0	2,0	2,0	2,5	3,0	2,5	15	Medium
P39	3,5	2,0	2,0	4,0	3,5	3,5	18,5	High
P40	2,0	1,0	1,0	2,0	1,5	1,5	9	Low
P41	2,5	2,0	2,0	2,5	3,0	3,0	15	Medium
P42	3,0	1,5	2,0	2,5	3,0	3,0	15	Medium
P43	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P44	3,5	2,0	2,0	2,5	2,5	2,5	15	Medium
P45	2,5	2,0	2,0	2,5	2,5	3,5	15	Medium
P46	3,0	2,5	2,0	2,5	2,5	2,5	15	Medium
P47	2,5	2,0	2,0	3,0	3,0	2,5	15	Medium
P48	2,0	1,0	1,0	1,5	1,5	2,0	9	Low
P49	3,5	2,0	2,0	4,0	4,0	3,0	18,5	High
P50	2,5	2,0	2,0	3,0	2,5	3,0	15	Medium
P51	3,5	2,5	2,0	3,5	3,0	3,0	17,5	High
P52	3,5	2,0	2,0	2,5	2,5	2,5	15	Medium
P53	3,0	2,0	2,0	2,5	2,5	3,0	15	Medium
P54	4,0	2,0	2,0	4,0	3,5	4,0	19,5	High
P55	2,0	2,0	2,0	3,0	3,0	3,0	15	Medium
P56	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P57	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P58	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P59	2,0	1,0	1,0	1,5	1,5	2,0	9	Low
P60	3,0	2,0	2,0	3,0	2,0	3,0	15	Medium
P61	3,5	2,0	2,0	4,0	3,0	3,5	18	High
P62	3,5	2,0	2,0	3,5	3,5	3,5	18	High
P63	2,5	2,0	2,0	3,0	3,0	2,5	15	Medium
P64	3,5	2,0	2,5	3,5	3,5	3,0	18	High
P65	2,5	2,0	2,0	3,0	3,0	2,5	15	Medium
P66	2,5	2,0	2,0	3,0	3,0	2,5	15	Medium
P67	4,0	2,0	2,0	4,0	3,5	4,0	19,5	High
P68	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P69	4,0	2,0	2,0	4,0	3,0	4,0	19	High
P70	3,5	2,0	2,0	3,5	4,0	4,0	19	High
P71	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P72	4,0	2,0	2,0	3,5	3,5	4,0	19	High
P73	3,5	2,0	2,0	4,0	3,5	3,0	18	High
P74	3,5	2,5	2,5	3,0	3,5	3,5	18,5	High
P75	3,5	2,0	2,0	4,0	3,5	4,0	19	High
P76	3,5	2,0	2,0	3,5	3,5	3,5	18	High
P77	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P78	2,0	2,0	2,0	3,0	3,0	3,0	15	Medium
P79	2,5	2,0	2,0	2,5	3,0	3,0	15	Medium
P80	4,0	2,0	2,0	4,0	4,0	4,0	20	High
P81	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P82	4,0	2,0	2,0	3,5	3,5	4,0	19	High
P83	4,0	2,0	2,0	3,5	3,0	3,5	18,5	High
P84	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P85	2,0	2,0	2,0	3,0	3,0	3,0	15	Medium
P86	2,0	2,0	2,0	2,5	3,0	3,0	14,5	Medium
P87	4,0	2,0	2,0	3,0	4,0	4,0	19	High
P88	4,0	2,0	2,0	3,5	3,5	4,0	19	High
P89	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P90	4,0	2,0	2,0	4,0	3,0	4,0	19	High
P91	3,5	2,0	2,0	3,5	3,5	3,5	18	High
P92	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P93	4,0	3,0	2,0	3,5	3,5	3,0	19	High
P94	4,0	2,0	2,0	3,5	3,5	3,5	18,5	High
P95	3,0	2,5	2,5	3,5	3,0	3,5	18,0	High
P96	4,0	3,0	2,0	3,5	3,5	3,0	19	High
P97	2,5	2,0	2,0	3,0	3,0	2,5	15	Medium
P98	3,0	2,0	2,0	3,0	3,0	2,5	15,5	Medium
P99	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P100	3,5	2,0	2,0	3,5	3,5	3,5	18	High
P101	4,0	2,0	2,0	3,5	4,0	4,0	19,5	High
P102	3,0	2,0	2,0	3,5	3,5	4,0	18	High
P103	3,0	2,0	2,5	3,5	3,5	3,5	18	High
P104	3,0	2,0	2,5	3,5	3,5	3,5	18	High
P105	3,0	2,0	2,5	3,5	3,5	3,5	18	High
P106	3,0	1,5	1,0	3,0	3,0	3,5	15	Medium
P107	3,5	2,0	1,5	4,0	4,0	4,0	19	High
P108	3,5	1,5	1,5	3,5	3,5	3,5	17	High
P109	3,0	2,0	2,0	2,0	2,5	3,0	14,5	Medium
P110	1,5	2,0	1,0	1,5	1,5	2,0	9,5	Low
P111	3,0	1,5	2,0	3,5	3,5	3,5	17	High
P112	2,5	2,0	1,5	3,0	3,0	3,0	15	Medium
P113	2,5	2,0	2,0	3,0	2,5	3,0	15	Medium
P114	2,5	2,0	2,0	2,0	3,0	3,5	15	Medium
P115	2,5	2,0	2,0	2,5	3,0	3,0	15	Medium
P116	4,0	2,0	2,0	4,0	4,0	3,5	19,5	High
P117	2,5	2,0	1,5	3,0	3,0	3,0	15	Medium
P118	2,5	2,0	1,5	2,5	3,5	2,5	14,5	Medium
P119	4,0	2,0	2,0	3,5	4,0	3,5	19	High
P120	4,0	2,0	2,0	4,0	4,0	3,0	19	High