

Review Article

Task Allocation in Distributed Software Development: A Systematic Literature Review

Marum Simão Filho ^{1,2} **Plácido R. Pinheiro** ¹ **Adriano B. Albuquerque** ¹
and **Joel J. P. C. Rodrigues** ^{3,4,5,6}

¹Graduate Program in Applied Informatics, University of Fortaleza, Brazil

²Information Systems, 7 de Setembro University Center, Brazil

³National Institute of Telecommunications (Inatel), Santa Rita do Sapucaí, MG, Brazil

⁴Instituto de Telecomunicações, Portugal

⁵ITMO University, St. Petersburg, Russia

⁶University of Fortaleza, Brazil

Correspondence should be addressed to Marum Simão Filho; marumsimao@gmail.com

Received 11 May 2018; Accepted 28 August 2018; Published 26 September 2018

Academic Editor: Giuseppe Zollo

Copyright © 2018 Marum Simão Filho et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Distributed development is a constant need in the software industry. The allocation of tasks, in turn, is a critical activity for any project, especially in a distributed setting. Assigning a task to an individual or team can be done in a rigorous manner, taking into account a number of factors. In this scenario, the more data on factors that influence task assignment are available to support the assignment decision, the more likely the project will succeed. As a result, companies have been looking to build their big data with historical project data, aiming to better their chances of success. Some studies about task allocation in distributed software development (DSD) have been carried out along the years aiming at mapping this topic and its features. Also, some approaches, models, and frameworks to task allocation in DSD have been proposed. This paper presents a systematic review of the literature whose objective is to identify works based on multicriteria models for assigning tasks in DDS projects, especially those based on qualitative decision-making methods, considering the cognitive validity present in some of these methods. The review results allowed identifying several interesting aspects and classifying the works according to a set of questions. In addition, it has made some important findings that may be very useful for those who intend to conduct research with qualitative multicriteria methods, since some of them employ techniques considered psychologically valid to address the complexity of the decision about the task allocation in distributed projects.

1. Introduction

The number of software projects grows day by day. The projects are related to several areas or fields and have different dimensions and characteristics. The number of software vendors grows at the same pace to meet this growing demand. As a result, the competition in the software market is quite high. Because of the competition, time requirements and quality requirements tend to be higher as well. Such a dynamic business environment requires that organizations develop, evolve, and maintain software at a much faster rate than in the past.

Large software companies often have offices in several different cities, or even, in various countries, to serve customers around the world. Teams in company branches can have different profiles or expertise. Some teams perform tasks that require proximity to the customer, but other tasks can be done remotely. Some kinds of activities can be carried out away from the large cities, where labour is often more expensive. Thus, companies take advantage of the low-cost workforce that is common when activities can be carried out in a distributed manner.

Software development consists of several different activities, such as requirement elicitation, analysis, and

design, coding, and testing, among others. Organizations that adopt consistent methodologies can distribute these activities in different sites, benefiting from the best skills of each remote team. Thus, Distributed Software Development (DSD) has become an increasingly present reality in modern companies. A critical challenge in distributed projects is related to the allocation of tasks to remote teams.

In this scenario, it is crucial that the company can be able to store as much data as possible about the history of its projects and all the underlying data, including the influencing factors. Consequently, more and more, companies have made efforts to build their big data ([1, 2]) with historical data of projects, aiming to improve their chances of success.

As we could realize, distributed development is a constant need in the software industry. The academy, in turn, has been investigating solutions to the many challenges of distributed software development. Thus, the literature provides some approaches that address the assignment of tasks in distributed software development settings.

This paper performs a systematic literature review on studies regarding approaches to task allocation in distributed software development. The goal was to explore the existing studies in search of any work related to qualitative criteria analysis, considering the cognitive validity present in some of these methods, according to Larichev [3]. Such a review differs from other investigations because we tried to analyze which of these approaches, processes, models, methods, and frameworks focus on task allocation in projects of distributed software development. Moreover, we looked for approaches that are based on qualitative multicriteria methods of decision-making, such as verbal decision analysis, since it employs psychologically valid techniques to address the complexity inherent in the task assignment decision in distributed projects. Other issues were also considered, such as support of automated tool, adaptability, flexibility, and the existence of a process or flowchart.

The remainder of the paper is organized as follows. Section 2 provides some of the main works related to the subject of this study found in the specialized literature. Section 3 outlines the adopted search method and explains how the review was planned and conducted. In Section 4, we present and discuss the review results. In Section 5, we report some limitations and threats to the validity of the research. Finally, Section 6 provides the conclusions and suggestions for future works.

2. Related Work

Some systematic literature reviews were found as result of our search and were considered very relevant to this research. We briefly describe some of them below.

Lamersdorf et al. [4] conducted a survey on the state of practice in DSD in which they investigated the criteria that influence task allocation decisions. Marques et al. [5] performed a systematic mapping, which enabled us to identify models that propose to solve the problems of allocation of tasks in DSD projects. They intended to propose a combinatorial optimization-based model involving classical task scheduling problems. Marques et al. [6] also performed

a tertiary review applying the systematic review method on systematic reviews that addressed DSD issues. The study revealed that most of the work focuses on summarizing the current knowledge about a particular research question, and some empirical studies are relatively small.

Galvina and Šmite [7] provided an extensive literature review for understanding the industrial practice of software development processes and concluded that the evidence of how these projects are organized is scarce. Despite that, they presented the models deducted from the development processes found in the selected literature and summarized the main challenges that may affect the project management processes.

In [8], Babar and Zahedi presented a literature review considering the studies published in the International Conference on Global Software Engineering (ICGSE) between 2007 and 2011. They found that the vast majority of the evaluated studies were in software development governance and its subcategories, and much of the work had focused on the human aspects of the GSD rather than technical aspects. There was also a growing trend to provide solutions to the problems identified in the area. However, many of the proposed solutions have not been rigorously evaluated. They also noted that while the GSD search production is increasing, there are significant gaps that must be filled, for example, to study the particular challenges and offer solutions for the different software development phases, such as design and testing.

da Silva et al. [9] developed another systematic literature review, but focusing on challenges and solutions in DSD project management. Finally, Simão Filho et al. [10] conducted a quasisystematic review of studies of task allocation in DSD projects that incorporate agile practices. The study brought together some other works, allowing identifying the many factors that influence the allocation of tasks in DSD.

We have noted that some of the models used multicriteria approaches, but usually focusing on quantitative and mathematical aspects. Moreover, most of the time, few factors were used to drive the allocation of tasks, such as hand labour costs, mainly. Risks and other relevant factors such as the workforce skills, innovation potential of different regions, or cultural factors were often insufficiently recognized [11].

The main difference between those reviews and this one is related to the research objectives. In this work, we intended to identify gaps in the set of existing approaches for task allocation in DSD that would allow us to drive some research to use multicriteria methods based on verbal decision analysis. The literature review is presented next.

3. Systematic Literature Review Planning and Conduction

Distributed development is a constant need in the software industry. The academy, in turn, has investigated solutions to the challenges of distributed software development. Thus, the literature presents some approaches to support the allocation of tasks in distributed software development environments. A systematic review of the

literature (SRL) was carried out to identify and characterize the existing approaches.

First, we wished to obtain studies related to the subject of this research. Then, we sought to carry out research for works with the focus on task allocation in distributed software development. In this research, we aimed to identify studies based on multicriteria models, especially in methods of verbal decision analysis. The next sections describe the research method adopted.

3.1. Objectives of the Review. The objective of this study was to analyze experience reports and scientific publications in the context of distributed software development, with the purpose of identifying the reported approaches and processes, as well as whether they were based on qualitative multicriteria models, mainly in methods of verbal decision analysis. We have established the following objectives:

- O1 Identify the existing approaches, processes, models, methods, and frameworks that support the allocation of tasks in distributed software development projects and analyze some characteristics.
- O2 Analyze which of these approaches, processes, models, methods, and frameworks are based on qualitative multicriteria methods of decision-making, especially on verbal decision analysis framework.
- O3 Identify factors or criteria that can be used to guide the decision on task allocation in distributed software development environments.

The target audience of this literature review was researchers who wish to get a view of the field of task allocation in DSD and the professionals who want to know the approaches, processes, models, methods, and frameworks that are available for use.

3.2. Research Method. According to Wazlawick [12], when a research is done on some technique or approach applied to a knowledge area, it is necessary to make a review on the approach itself, on the application area, and mainly on applications that have already been attempted with this approach (or similar approaches) in the same area or in related areas. A systematic review of the literature (or merely systematic review) is a means of identifying, evaluating, and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest [13]. According to Kitchenham and Charters [13], systematic reviews are suitable to identifying any gaps in the current research to suggest areas for further investigation. Moreover, a systematic review is a reliable, rigorous, and auditable research method, which tends to produce results that are more reliable if compared to an informal review.

Kitchenham [14] structures the stages of a systematic review in three main phases: planning the review, conducting the review, and discussing the review. The systematic review process applied in this research was inspired by this roadmap, whose phases are detailed below (see Table 1).

TABLE 1: Systematic review process applied in this study.

Phase	Stage/activity
Planning the review (3.3)	Specify research questions (3.3.1)
	Define search strategy (3.3.2)
	Select publications (3.3.3)
	Define search scope (3.3.4)
	Select sources (3.3.5)
	Define inclusion and exclusion criteria (3.3.6)
Conducting the review (3.4)	Collect data (3.4.1)
	Present data (3.4.2)
Discussing the results (4)	Questions analysis (4.1)
	Main conclusions (4.2)

3.3. Planning the Review. Specifying research questions is the most important part of any systematic review [13]. Thus, the first step in research planning is to define the research question. To obtain the information desired for this study, we have identified research questions associated with each objective.

3.3.1. Specifying Research Questions. The central research questions (primary questions (PQ)) were defined as follows:

- PQ1 What approach was used or suggested for the allocation of tasks in distributed software development projects? The purpose of this question was to identify which approaches, processes, models, methods, or frameworks are present in the literature that assists in the allocation of tasks in distributed software development projects, either automatically or manually.
- PQ2 Is the approach based on qualitative multicriteria methods of decision-making? This question is aimed at identifying works related to this research, that is, which studies described approaches that support the allocation of tasks in distributed environments and were based on qualitative multicriteria methods of decision-making.
- PQ3 Is the approach based on any method of verbal decision analysis framework? With this question, we intended to determine whether there is any approach directly linked to the verbal decision analysis methods.
- PQ4 Does the study mention any factor or criterion that influences the allocation of tasks in distributed software environments? With this question, we intended to identify which factors or criteria were used to guide the decision on task allocation in distributed software development environments.

Additionally, the study is aimed at answering some secondary research questions (SQ):

- SQ1 Does the approach describe a process? This question is aimed at identifying if the approach describes some process or flow that guides the execution of the activities in order to reach the proposed objectives.
- SQ2 Does the approach have any automated support tool? This question had the purpose of identifying whether the approach uses or implemented some automated tool to support the execution of activities.
- SQ3 Does the approach apply influencing factors extracted from the specific literature? The purpose of this question was to identify whether the approach used or applied influencing factors found in the literature on distributed software development or whether it created its list of factors.
- SQ4 Can the approach be instantiated from an organization standard? The objective of this question was to investigate whether the approach could be instantiated from an organization standard; that is, once the company has a template or even some templates, we wanted to know if it would be possible to create instances from these templates by changing only some data (parameters or factors or others).
- SQ5 Does the approach reuse knowledge? With this question, we sought to know if the approach allowed the use of knowledge from other sources (factors, criteria, relations, etc.) or even from the company's past.
- SQ6 Does the approach presuppose consensus meeting? This question is aimed at finding out if the approach included some activity or stage dedicated to evaluate the generated results and decide on their application.
- SQ7 Does the approach presuppose feedback of results? With this question, we wanted to know if it was possible to feedback the approach with generated results, allowing model calibration.
- SQ8 Does the approach allow using a subset of the whole process? The purpose of this question was to find out whether the approach could only be fully applied or whether it was possible to apply only a few stages of the process, taking advantage of the knowledge gained in other experiences or the company's past.
- SQ9 Does the approach allow modifying the main parameters (allocation object and executing unit)? With this question, we intended to know how parametrizable the approach was, that is, whether it was possible to apply it to different task sizes (individual tasks, task packages, stages of software development, etc.) as well as for different levels of executing units (individuals, teams, departments, offices, etc.).
- SQ10 Has the approach been applied in the real-world situations? With this question, we wanted to know if the approach was submitted to some experience of use in the real context of the software industry, i.e., if the approach was experienced in practice, either partially or totally.
- SQ11 Has the approach been evaluated by users or professionals? The objective of this question was to know the rigor with which the presented approach was submitted for validation or evaluation by users or professionals who participated in some kind of experience of use or similar activity, allowing knowing the level of maturity of the proposed approach.
- SQ12 Has the approach been evaluated by knowledge area experts? The purpose of this question is to know the rigor with which the presented approach was submitted for validation or evaluation by professional or researchers with a high degree of experience and knowledge in their areas of practice, allowing knowing the level of maturity of the proposed approach.
- 3.3.2. Defining Search Strategy.* It was necessary to determine and follow a search strategy [13]. This was important so that relevant studies were included in the search results. To achieve broad coverage in the search, we aimed to carry out the research by making several combinations of terms related to the subject of the research, both in digital libraries of the academic area and in search engines where it was possible to obtain related works.
- (1) Search Method.* Automated searches were performed on electronic database engines or digital libraries using search expressions, to reach a large number of publications. Manual searches were avoided because the amount of work is usually high. Also, we ran the risk of leaving out relevant works. Thus, to define the search, we established the following strategy:
- (i) Derive the main terms from the research questions
 - (ii) Identify synonyms, plurals, and related terms
 - (iii) Use the OR logical operator to incorporate the synonyms
 - (iv) Use the AND logic operator to connect the parameters
 - (v) Check the terms in the titles, abstracts, and keywords of the publications
- (2) Search String.* It was necessary to establish the population, intervention, comparison, and production parameters to formulate the search sequence, based on research questions, as described in [13] and shown in Table 2.
- The element of comparison was discarded. Thus, the search string consisted of three parts: (population) AND

TABLE 2: Question structure.

Population	Works related to distributed software development.
Intervention	Works that addressed task allocation.
Comparison	It did not apply, as it is not the objective of this work to carry out comparative analyses.
Outcomes	Approaches, processes, or other kinds of solutions that addressed the allocation of tasks in distributed software development.

TABLE 3: Parameters of search question.

Population	distributed software development OR global software development OR collaborative software development OR global software engineering OR globally distributed work OR collaborative software engineering OR distributed development OR distributed team OR global software teams OR globally distributed development OR geographically distributed software development OR offshore software development OR offshoring OR offshore OR offshore outsourcing OR dispersed teams
Intervention	task allocation OR task assignment OR task distribution
Outcomes	approach OR process OR model OR method OR framework

(intervention) AND (outcomes). Different authors can use distinct terminologies to represent the same subject. For this reason, it is common to create a broader expression involving several synonyms of the searched term to guarantee the presence of a greater number of related works, even if this can significantly increase the amount of returned work. The synonyms were then connected through the logical OR operator to form the search string. Thus, we defined the following strings (see Table 3).

The complete search string was defined as follows:

(distributed software development OR global software development OR collaborative software development OR global software engineering OR globally distributed work OR collaborative software engineering OR distributed development OR distributed team OR global software teams OR globally distributed development OR geographically distributed software development OR offshore software development OR offshoring OR offshore OR offshore outsourcing OR dispersed teams) AND (task allocation OR task assignment OR task distribution) AND (approach OR process OR model OR method OR framework)

(3) *Protocol Experiment*. We performed a pilot experiment that allowed us to identify some relevant works to the theme of our research. For this, the following studies were considered for testing the search string: Lamersdorf and Münch [15], Almeida and Albuquerque [16], and Prikladnicki et al. [17].

3.3.3. *Selecting Publications*. The selection of the publications was structured in three stages: (i) preliminary selection, (ii) primary selection, and (iii) final selection. In the first stage, we selected the digital libraries, performed the search expression for each base, and compiled all the returned works, along with other works identified in manual searches (other literature reviews and theses). In the second stage, we discarded duplicate works and nonscientific publications, based on the full reference of each work. Also in this step, we applied the first selection filter, aiming to analyze the works abstract according to the inclusion and exclusion

criteria. At the end of this step, we had a list of candidate publications. In the last stage, we downloaded the candidate publications, which were read and analyzed again according to the inclusion and exclusion criteria, corresponding to the second selection filter. Finally, the research questions were applied to these works and the results were consolidated.

3.3.4. *Defining Search Scope*. We considered three dimensions for the scope of our search: time period, language, and source. The period of publication of the papers was not limited so that the research was comprehensive, including works that could be sources of reference in the study on the topic explored. Thus, we take into account the works published until the date of the research, i.e., the first quarter of 2016.

Regarding the language, we decided to use the English and Portuguese languages. The choice of English language is due to its adoption by most international conferences and journals related to the research topic and because most relevant publishers use it. The choice of the Portuguese language is due to its adoption by major national conferences and journals of Software Engineering, especially the ones supported by the Brazilian Computer Society, and because there are many master dissertations and doctoral thesis of main Brazilian universities that were written in Portuguese.

3.3.5. *Selecting Sources*. Concerning sources of the work, the search should be carried out in digital libraries through its search engines. Therefore, to be selected, libraries should meet the following criteria:

- (i) Have search engines that allow the use of logical expressions
- (ii) Allow search in specific fields of publications, such as title, abstract, and keywords
- (iii) Belong to one of the publishers listed in the Portal of Periodicals of Brazilian Coordination for the

Improvement of Higher Level-or Education-Personnel (CAPES)

- (iv) Contain publications from the area of computing at their base

Thus, we selected the following digital libraries:

- (i) ACM Digital Library (<http://dl.acm.org>)
- (ii) IEEE Xplore Digital Library (<http://ieeexplore.ieee.org>)
- (iii) Elsevier Scopus (<http://www.scopus.com>)
- (iv) ScienceDirect (<http://www.sciencedirect.com>)

Also, theses and dissertations related to the research theme were also considered. These works were obtained through manual searches using the search engine provided by Google Scholar (<http://scholar.google.com>).

3.3.6. Defining Inclusion and Exclusion Criteria. Study selection criteria were intended to identify those primary studies that provided direct evidence about the research question [13]. Thus, we have adopted some inclusion and exclusion criteria to ensure that only scientific papers that met the objectives of the review were analyzed. Therefore, for some publication to be analyzed in this study, it would need to meet the following inclusion criterion: studies that describe approaches that support the allocation of tasks in distributed software development context.

Exclusion criteria were also established in order to discard the publications returned by searches that did not belong to the scope of the survey: (1) the study did not meet the inclusion criterion; (2) the study was unavailable for download without cost; and (3) the study was unavailable in English or Portuguese.

All the studies returned in the searches were analyzed considering the established criteria. First, we examined title, keywords, and abstract of each publication by checking the inclusion and exclusion criteria (first filter). Papers that met some exclusion criteria were removed from the list of candidate publications. Afterwards, the remaining publications were downloaded and entirely read, reevaluating the inclusion and exclusion criteria (second filter). In the end, the remaining papers composed the final list.

3.4. Conducting the Review

3.4.1. Collecting Data. The systematic review was developed in the first quarter of 2016 and was initially conducted by only one researcher. As planned, the selection procedure was carried out in three steps: preliminary selection, primary selection, and final selection. In the preliminary selection stage, the search string was applied to the chosen digital libraries, resulting in 415 publications returned. We then proceeded to the second stage, primary selection, in which the selection filters were applied; that is, we analyzed titles, keywords, and abstracts verifying the inclusion and exclusion criteria, resulting in 63 candidate publications. In the third stage, final selection, we read the full text of the candidate

publications, again verifying the inclusion and exclusion criteria, resulting in 21 publications. The selected publications were presented to a second researcher, who agreed with the final selection. Figure 1 illustrates the process and presents its findings.

3.4.2. Presenting Data. After the selection stage, we began to analyze the publications. In this step, each paper was examined to understand the described approach, as well as to answer the research questions, both primary and secondary questions. Table 4 indicates the codes used in data consolidation table. Table 5 provides an overview of the selected and consolidated papers as well as the answers to the research questions. Cells marked with “✓” indicate that a positive answer to the corresponding question was found in the investigated works. Cells marked with “—” indicate that no conclusive information about the research question was found in the consulted publications. It is important to emphasize that this analysis considered the information that was available and accessible during the period of accomplishment of this work, as well as the author’s understanding of issues that were not explained explicitly in the texts read.

4. Results and Discussion

4.1. Question Analysis. In this subsection, question responses are analyzed and discussed: firstly, the primary questions, then, the secondary ones.

- PQ1 What approach was used or suggested for the allocation of tasks in distributed software development projects?

Eleven different approaches were identified in the consulted publications. The researched approaches used different denominations, being six models, two frameworks, one reference model, one environment, and one tool. Although the approaches are focused on the same central theme, that is, allocation of tasks in a distributed scenario, some of them tackle specific problems, such as 24-hour development [30], allocation in SCRUM projects [16], or component allocation in Distributed Software Product Lines [35]. The methods and techniques applied to solve the task allocation problem in DSD were also quite varied. Some approaches were based on multicriteria methods ([16, 34]), one tool used Fuzzy Logic [23], one model adopted simulation techniques [27], and another based on the Bokhari algorithm and Bayesian networks [15]. Also, some approaches have been devoted to indicating the best recommendation for the allocation (e.g., [15, 35]), while others have proposed reference models and processes for the allocation flow of tasks (e.g., [17, 19]).

- PQ2 Is the approach based on qualitative multicriteria methods of decision-making?

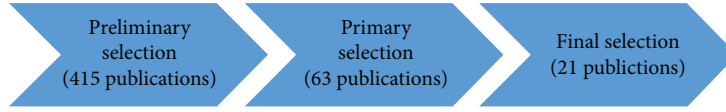


FIGURE 1: Publication selection process.

TABLE 4: Codes used in the data consolidation table.

Question ID	Description
PQ1	What approach was used or suggested for the allocation of tasks in distributed software development projects?
PQ2	Is the approach based on qualitative multicriteria methods of decision-making?
PQ3	Is the approach based on any method of verbal decision analysis framework?
PQ4	Does the study mention any factor or criterion that influences the allocation of tasks in distributed software environments?
SQ1	Does the approach describe a process?
SQ2	Does the approach have any automated support tool?
SQ3	Does the approach apply influencing factors extracted from the specific literature?
SQ4	Can the approach be instantiated from an organization standard?
SQ5	Does the approach reuse knowledge?
SQ6	Does the approach presuppose consensus meeting?
SQ7	Does the approach presuppose feedback of results?
SQ8	Does the approach allow using a subset of the whole process?
SQ9	Does the approach allow modifying the main parameters (allocation object and executing unit)?
SQ10	Has the approach been applied in real-world situations?
SQ11	Has the approach been evaluated by users or professionals?
SQ12	Has the approach been evaluated by knowledge area experts?

We have found only two approaches that were structured in qualitative multicriteria methods of verbal decision analysis: McSDS [16] and the model of Barcus and Montibeller [34]. In fact, Barcus and Montibeller [34] reinforce the maturity that MCDA has achieved, with well-developed methods and many real-world interventions. However, they believe that more research focused on the process of supporting groups of decision-makers with MCDA is needed.

PQ3 Is the approach based on any method of the verbal decision analysis framework?

No approach based on verbal decision analysis methods was found in the searched literature. This situation reveals a gap in the bibliographic studies. Once task allocation in the DSD environment consists of a multicriteria decision-making problem and involves a certain degree of subjectivity, it would be useful to investigate how the VDA framework could be applied in this context, since VDA proposes to deal with problems with many criteria through qualitative analysis.

PQ4 Does the study mention any factor or criterion that influences the allocation of tasks in distributed software environments?

All investigated approaches have taken into account several factors, from personal aspects,

such as technical skills and project manager maturity, to geographic issues, such as cultural and time zone differences. Some factors appeared more frequently, such as team skills and labour cost. Barcus and Montibeller [34] dealt not only with software engineering issues but also with “soft” and strategic issues, like team satisfaction and training opportunities. Different classifications have also been found. Lamersdorf and Münch [15] classified the factors (or criteria) for task allocation in standard and custom software development. Prikladnicki [17] identified five categories of factors: development process, project, stakeholders, organization, and dispersion. Ruano-Mayoral et al. [19], in turn, defined metrics for the factors used in their framework.

SQ1 Does the approach describe a process?

A process is necessary to guide users in performing the activities to achieve the proposed goal. Nevertheless, only a few studies have presented approaches that describe some process or flow of activities: TAMRI [15], Ruano-Mayoral’s Methodological Framework [19], MuNDDoS [17], MCDA model of Barcus and Montibeller [34], and the Recommendation Framework for SPL [35].

SQ2 Does the approach have any automated support tool?

TABLE 5: Mapping of approaches and research questions.

Approach	PQ1	PQ2	PQ3	PQ4	SQ1	SQ2	SQ3	SQ4	SQ5	SQ6	SQ7	SQ8	SQ9	SQ10	SQ11	SQ12
TAMRI [15, 18]	Based on multiple criteria and influencing factors to support the systematic decision of task allocation in distributed development projects.	—	—	✓	✓	✓	✓	✓	✓	—	—	✓	✓	✓	✓	✓
Ruano-Mayoral's Methodological Framework [19, 20]	A methodological framework to allocate work packages among participants in global software development projects.	—	—	✓	✓	✓	✓	✓	—	—	—	—	✓	✓	✓	✓
McDSDS [16, 21, 22]	A multicriteria decision model for planning distributed software development projects with Scrum	✓	—	✓	—	✓	✓	—	—	—	—	—	—	✓	—	—
DIMANAGER [23–26]	A tool inserted in the context of DiSEN environment to support the necessary selection of human resources based on the Fuzzy Logic.	—	—	✓	—	✓	✓	—	—	—	—	—	—	✓	✓	✓
GSD Simulation Model [27–29]	A model that allows capturing dynamic and discrete aspects of a distributed software development environment and provides information related to productivity, allocated resources, communication, and coordination during the project.	—	—	✓	—	—	✓	✓	✓	—	✓	✓	✓	—	—	—
24-Hour Development Model [30]	It uses a DAG (directed acyclic graph) and is based on the critical path method. The model performs the allocation seeking to reduce the duration of the project.	—	—	✓	—	—	✓	✓	—	—	—	—	✓	✓	—	—
MuNDDoS [17, 31]	A reference model for distributed software development, which contains a project allocation process that involves the selection of projects to be developed in each distributed unit, according to an allocation policy defined by the organization.	—	—	✓	✓	—	✓	✓	✓	✓	✓	✓	✓	—	—	—
Global Studio Project [32]	A student-based software development project that has been instrumented for data collection to support empirical studies on communication, coordination, and collaboration among distributed teams.	—	—	✓	—	✓	✓	—	—	—	—	✓	✓	—	—	—
NextMove [33]	A model that combines project management techniques and object-oriented process modeling to solve the problem of coordination and allocation of tasks for distributed teams in agile development context. Task allocation is based on the AHP approach.	—	—	✓	—	✓	✓	✓	✓	—	✓	—	—	—	—	—
Barcus and Montibeller [34]	A multicriteria model for supporting the distributed teamwork allocation decision for a major global software company	✓	—	✓	✓	—	✓	✓	✓	—	✓	—	✓	—	—	✓
Recommendation Framework for allocation of dev teams in dist. proj. SPL [35]	A framework that aims at providing recommendations on how to allocate teams to software components in distributed Software Product Lines projects by considering technical and nontechnical aspects.	—	—	✓	✓	—	✓	✓	✓	—	✓	✓	✓	✓	—	—

Automated support tools are helpful because they make it easier to use the approach, reducing the effort to perform the process or activity. Besides, other benefits such as storing historical data and speed in generating results are quite useful for these kinds of approaches, which often require knowledge reuse and rapid decision-making. For this analysis, we considered tools that have been implemented for the approach or third-party tools that were used by the approach. We did not consider spreadsheets in Excel. Regarding automated support, more than half of the evaluated approaches provided some kind of tool, namely, TAMRI [15], Ruano-Mayoral's Methodological Framework [19], McDSDS [16], DIMANAGER [23], Global Studio Project [32], and NextMove [33].

SQ3 Does the approach apply influencing factors extracted from the specific literature?

The literature mentions a good number of factors that influence the assignment of tasks in distributed projects. All the consulted approaches make use of these factors present in the literature in their mechanisms or processes.

SQ4 Can the approach be instantiated from an organization standard?

For some approaches, it is possible to establish a standard within the organization. With this, every time a decision on task allocation needs to be made, the approach can be instantiated from the existing standard using current or historical data, and the lessons learned. This facilitates the process as it avoids the overhead of having to start the entire decision process without any data. Regarding this, in several approaches it was possible to find evidence that it was possible to create instances from an organizational standard, such as TAMRI [15], Ruano-Mayoral's Methodological Framework [19], GSD Simulation Model [27], 24-Hour Development Model [30], MuNDDoS [17], NextMove [33], MCDA model of Barcus and Montibeller [34], and Recommendation Framework for SPL [35].

SQ5 Does the approach reuse knowledge?

Decision-making processes use and generate a lot of information. Historical data, records of decisions, lessons learned, all these are examples of knowledge generated during decision-making. Storing such knowledge, either automatically or manually, and using it as input in future situations can increase the chances of achieving better results. Thus, the following approaches provided indications that it was possible to reuse knowledge: TAMRI [15], GSD Simulation Model [27], MuNDDoS [17], NextMove [33], MCDA model of Barcus and Montibeller [34], and Recommendation Framework for SPL [35].

SQ6 Does the approach presuppose consensus meeting?

After executing any approach or model and generating the allocation recommendations, it is important to assess the feasibility of implementing such recommendations, as external factors or issues not initially thought may arise. In such cases, holding consensus meetings appears to be a good practice, as they allow assessing such situations. Despite this, only the reference model of MuNDDoS [17] mentions something like a consensus meeting.

SQ7 Does the approach presuppose feedback of results?

In the decision-making process, recording the success or failure of a decision, as well as the involved aspects, is very relevant to calibrate the model. For this, it is necessary to feedback the model with the results produced in previous iterations. Thus, the incorporated knowledge can be applied to new situations. Concerning this, the following approaches mentioned feeding back the model with some kind of resulting data: GSD Simulation Model [27], MuNDDoS [17], NextMove [33], MCDA model of Barcus and Montibeller [34], and Recommendation Framework for SPL [35].

SQ8 Does the approach allow using a subset of the whole process?

Models composed of various activities or modules may allow applying only a subset of the whole, without compromising the quality of the outcome. This is usually possible when there is a knowledge base that allows starting the process from the intermediate steps. In this regard, the following models provide evidence that it is possible to perform only a subset of activities or steps of the full version: TAMRI [15], GSD Simulation Model [27], MuNDDoS [17], Global Studio Project [32], and Recommendation Framework for SPL [35].

SQ9 Does the approach allow modifying the main parameters (allocation object and executing unit)?

A model is created to meet a demand or need, thus solving a particular problem. However, it may be possible to adapt or extend the model to suit other scenarios or dimensions. For example, an approach initially designed to support task allocation to team members can be adapted to support the allocation of work packages to teams. In this sense, most of the surveyed models presented indications that they can be adapted in relation to the allocation object and the executing unit, except McDSDS [16], DIMANAGER [23], and NextMove [33].

SQ10 Has the approach been applied in real-world situations?

Applying an approach in real-world scenarios is essential for its validation, calibration, and improvement. Regarding this, the following papers have reported on the application in real situations: TAMRI [15], Ruano-Mayoral's Methodological Framework [19], McDSDS [16], DIMANAGER [23], 24-Hour Development Model [30], and Recommendation Framework for SPL [35].

SQ11 Has the approach been evaluated by users or professionals?

In addition to applying the approach in real-world scenarios, it is also quite pertinent to conduct some evaluation in the process, whether quantitative or qualitative. The purpose of the evaluation is to check whether the objectives of the approach have been achieved or not, identifying weaknesses and strengths, as well as opportunities for improvement. Concerning this question, the following approaches mentioned having performed some evaluation by users or professionals participating in the experiences of use of the approach: TAMRI [15], Ruano-Mayoral's Methodological Framework [19], and DIMANAGER [23].

SQ12 Has the approach been evaluated by knowledge area experts?

Also regarding the evaluation, it is a common practice to submit a model to the evaluation of experts in the field of knowledge. In this sense, the following works indicated having applied some evaluation with project management experts: TAMRI [15], Ruano-Mayoral's Methodological Framework [19], DIMANAGER [23], and MCDA model of Barcus and Montibeller [34].

4.2. Main Review Conclusions. The approaches were given different denominations, such as process, framework, and model, among others, which required more attention in search of publications. All approaches have taken into account several factors, from personal aspects, such as technical skills and project manager maturity, to geographic issues, such as proximity to the client and time zone differences. Some factors appeared more frequently, such as team skills and labour cost.

We have noted some variation in the granularity of the task to be allocated. Some approaches consider individual tasks, such as [25, 30, 33, 34], while others focus on groups or packages of tasks ([20, 34]), or even entire projects, such as [17]. We also have perceived variations concerning the destination of the allocation. Some studies addressed the assignment of tasks to individuals [25, 30, 33]; others consider teams, offices, or branches ([11, 16, 17, 27, 34]). Some studies defined conceptual models, such as frameworks and reference models ([17, 19]), while others implemented automated tools from the proposed models ([23]). It is important to note that, considering the papers analyzed, only three presented approaches based on multicriteria decision-making methods ([16, 33, 34]). However, only two of them

TABLE 6: Total of positively answered questions.

Approach	Count
TAMRI	11
Ruano-Mayoral's Methodological Framework	9
Recommendation Framework for allocation of dev teams in dist. proj. SPL	9
MuNDDoS	9
Barcus and Montibeller	9
GSD Simulation Model	7
NextMove	6
DIMANAGER	6
McDSDS	5
Global Studio Project	5
24-Hour Development Model	5

were based on qualitative analysis, ([16, 34]), and none were based on methods of the verbal decision analysis framework. In addition to works listed above, other studies did not describe approaches but did some mention of factors or criteria that influence or affect the allocation of tasks in distributed software development. Among them, we highlight [36–42].

Considering the 15 objective questions assessed in this study (three primary questions (PQ2 to PQ4) and twelve secondary questions), TAMRI was the best evaluated approach, which scored in 11 out of 15 questions. For the criteria considered in this study, the McDSDS, Global Studio Project, and 24-Hour Development Model approaches obtained the lowest score (only 5 out of 15 questions). The PQ4 and SP3 criteria, both related to the influencing factors for task allocation in DSD, were successful by all approaches, while only one approach addressed the SQ6 question (consensus meeting) and no approach satisfied the P3 question (based on any VDA method). Tables 6 and 7 detail the performance of the approaches researched and the issues investigated, respectively.

Other studies have conducted systematic reviews or extensive systematic mapping on task allocation in distributed software development environments, such as [4–10, 41, 43–47]. As mentioned previously, the main difference between the other reviews and this one is related to the research objectives. In this work, we intended to identify gaps in the set of existing approaches for task allocation in DSD that would allow us to drive some research to use multicriteria methods based on verbal decision analysis. Since no such approach was found, we understood that our goal had been achieved in this sense. Moreover, many influencing factors have been identified in the DSD task allocation decision process, which may be useful for future works. Also, several aspects were analyzed in the existing approaches that allowed us to delineate some desirable characteristics for a new approach in this field. Finally, this review served as support for the research developed in other works such as [48–54].

TABLE 7: Total approaches that scored for each question.

Q. ID	Description	Count
PQ4	Does the study mention any factor or criterion that influences the allocation of tasks in distributed software environments?	11
SQ3	Does the approach apply influencing factors extracted from the specific literature?	11
SQ4	Can the approach be instantiated from an organization standard?	8
SQ9	Does the approach allow modifying the main parameters (allocation object and executing unit)?	8
SQ2	Does the approach have any automated support tool?	6
SQ5	Does the approach reuse knowledge?	6
SQ10	Has the approach been applied in real-world situations?	6
SQ1	Does the approach describe a process?	5
SQ7	Does the approach presuppose feedback of results?	5
SQ8	Does the approach allow using a subset of the whole process?	5
SQ12	Has the approach been evaluated by knowledge area experts?	4
SQ11	Has the approach been evaluated by users or professionals?	3
PQ2	Is the approach based on qualitative multicriteria methods of decision-making?	2
SQ6	Does the approach presuppose consensus meeting?	1
PQ3	Is the approach based on any method of verbal decision analysis framework?	0

5. Limitations

The main threat concerns the publication bias to which all studies of this type are subject, i.e., most literature reviews do not clarify the underlying justification.

About sources, we do not access all possible sources. Also, there were issues related to restricted access to some digital resources. We used only reputed digital resources that were accessible to us. However, we understand that we have covered enough literature to generalize our findings. It is important to emphasize that the publication analysis took into account the information that was available and accessible during the period of accomplishment of this work, as well as the author's interpretation of issues that were not provided explicitly in the texts read. To mitigate such a threat, the results analysis was discussed with two other researchers who also had access to the selected publications.

Regarding the interface of the search engines, it was not possible to use the same search string in all digital libraries. Some of them made it possible to formulate more complex queries, while others provided only a few filters. Another limitation identified was the lack of relevant information for some works researched, such as the location where the research was conducted and the time in which the research was developed, as well as the number and profile of the researchers involved.

More detailed information on the limitations and threats to the validity of this study can be found in [48].

6. Conclusion and Future Work

This paper described a systematic review of the literature on studies related to approaches to the allocation of tasks in distributed software development. The goal was to evaluate the existing studies in search of any work related to qualitative multicriteria analysis. The results showed that, although it is a subject present in recent publications and others not so

recent, few studies have some relation to task allocation using qualitative multicriteria methods. If we consider methods of the verbal decision analysis framework, no work has been identified.

The research carried out allowed a better understanding of the main approaches related to the allocation of tasks in distributed software projects. Moreover, the results obtained in the systematic review allowed us to identify some research opportunities such as applying qualitative multicriteria methods for task allocation in DSD projects. The review also allowed identifying numerous factors or criteria that influence or affect the allocation of tasks in distributed software development.

As future works we suggest (i) to investigate the impact of the influencing factors identified in this review on task allocation in both centralized and distributed development projects, (ii) to explore the application of qualitative multicriteria-based approaches to the problem of task assignment in DSD scenario, and, finally, (iii) to work on a proposal of an approach based on verbal decision analysis to support the allocation of tasks in DSD projects.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

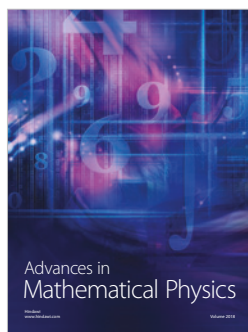
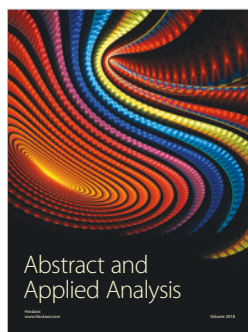
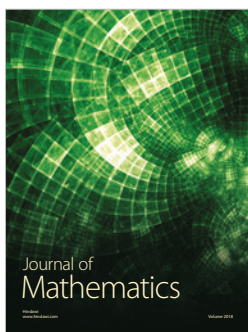
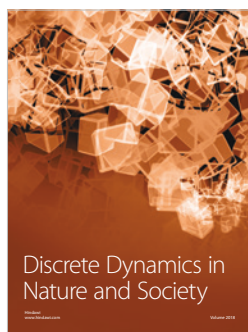
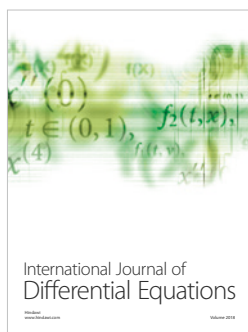
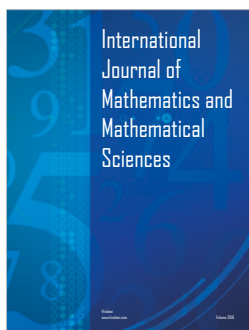
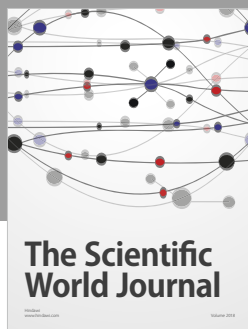
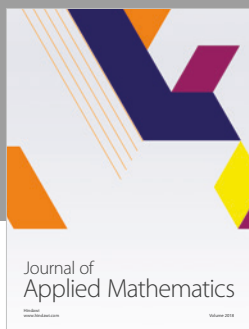
The first author is thankful for the support given by the 7 de Setembro University Center during this project. The second author is grateful to the National Council for Scientific and Technological Development (CNPq) via Grant #305844/2011-3. The fourth author is grateful to the national funding from the FCT - *Fundação para a Ciência e a Tecnologia* through the UID/EEA/500008/2013 Project; by the Government of the Russian Federation, Grant 08-08; by FINEP, with

resources from FUNTEL, Grant no. 01.14.0231.00, under the Radiocommunication Reference Center (*Centro de Referência em Radiocomunicações - CRR*) project of the National Institute of Telecommunications (*Instituto Nacional de Telecomunicações - Inatel*), Brazil; and by the Brazilian National Council for Research and Development (CNPq) via Grant No. 309335/2017-5. The authors would like to thank the Edson Queiroz Foundation/University of Fortaleza for all the support.

References

- [1] S. Sagioglu and D. Sinanc, "Big data: a review," in *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pp. 42–47, San Diego, CA, USA, 2013.
- [2] A. Katal, M. Wazid, and R. H. Goudar, "Big data: issues, challenges, tools and good practices," in *2013 Sixth International Conference on Contemporary Computing (IC3)*, pp. 404–409, Noida, India, 2013.
- [3] O. I. Larichev, "Cognitive validity in design of decision-aiding techniques," *Journal of Multi-Criteria Decision Analysis*, vol. 1, no. 3, pp. 127–138, 1992.
- [4] A. Lamersdorf, J. Munch, and D. Rombach, "A survey on the state of the practice in distributed software development: criteria for task allocation," in *2009 Fourth IEEE International Conference on Global Software Engineering*, pp. 41–50, Limerick, Ireland, 2009.
- [5] A. B. Marques, R. Rodrigues, R. Prikladnicki, and T. Conte, "Alocação de Tarefas em Projetos de Desenvolvimento Distribuído de Software: Análise das Soluções Existentes," in *V Brazilian Workshop on Distributed Software Development*, São Paulo, Brazil, 2016.
- [6] A. B. Marques, R. Rodrigues, and T. Conte, "Systematic literature reviews in distributed software development: a tertiary study," in *2012 IEEE Seventh International Conference on Global Software Engineering*, pp. 134–143, Porto Alegre, Brazil, 2012.
- [7] Z. Galviņa and D. Šmite, *Software Development Processes in Globally Distributed Environment*, Datorzinātne un informācijas Teh, 2011.
- [8] M. A. Babar and M. Zahedi, *Global Software Development : a Review of the State-Of-the-Art (2007-2011)*, IT University Technical Report Series, 2012.
- [9] F. Q. B. da Silva, C. Costa, A. C. C. França, and R. Prikladnicki, "Challenges and solutions in distributed software development project management: a systematic literature review," in *2010 5th IEEE International Conference on Global Software Engineering*, pp. 87–96, Princeton, NJ, USA, 2010.
- [10] M. Simão Filho, P. R. Pinheiro, and A. B. Albuquerque, "Task allocation approaches in distributed agile software development: a quasi-systematic review," in *Software Engineering in Intelligent Systems*, vol. 349 of *Advances in Intelligent Systems and Computing*, pp. 243–252, Springer, Cham, 2015.
- [11] A. Lamersdorf, J. Münch, and D. Rombach, "Towards a multi-criteria development distribution model: an analysis of existing task distribution approaches," in *2008 IEEE International Conference on Global Software Engineering*, pp. 109–118, Bangalore, India, 2008.
- [12] R. Wazlawick, *Metodologia de pesquisa para ciência da computação, 2a edição*, Elsevier, Brazil, 2nd edition, 2014.
- [13] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature reviews in Software Engineering Version 2.3," *Engineering*, vol. 45, p. 1051, 2007.
- [14] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, pp. 1–26, 2004.
- [15] A. Lamersdorf and J. Münch, "A multi-criteria distribution model for global software development projects," *Journal of the Brazilian Computer Society*, vol. 16, no. 2, pp. 97–115, 2010.
- [16] L. Almeida and A. Albuquerque, "A multi-criteria model for planning and fine-tuning distributed Scrum projects," in *2011 IEEE Sixth International Conference on Global Software Engineering*, pp. 75–83, Helsinki, Finland, 2011.
- [17] R. Prikladnicki, S. Marczak, and J. N. Audy, "MuNDDoS: a research group on global software development," in *2006 IEEE International Conference on Global Software Engineering (ICGSE'06)*, pp. 251–252, Florianopolis, Brazil, 2006.
- [18] A. Lamersdorf and J. Munch, "TAMRI: a tool for supporting task distribution in global software development projects," in *2009 Fourth IEEE International Conference on Global Software Engineering*, pp. 322–327, Limerick, Ireland, 2009.
- [19] M. Ruano-Mayoral, R. Colomo-Palacios, J. M. Fernández-González, and Á. García-Crespo, "Towards a framework for work package allocation for GSD," in *On the Move to Meaningful Internet Systems: OTM 2011 Workshops*, pp. 200–207, Springer, Berlin, Heidelberg, 2011.
- [20] M. Ruano-Mayoral, C. Casado-Lumbreras, H. Garbarino-Alberti, and S. Misra, "Methodological framework for the allocation of work packages in global software development," *Journal of Software: Evolution and Process*, vol. 26, no. 5, pp. 476–487, 2014.
- [21] L. H. Almeida, P. R. Pinheiro, and A. B. Albuquerque, "Applying multi-criteria decision analysis to global software development with scrum project planning," in *Rough Sets and Knowledge Technology*, pp. 311–320, Springer, Berlin, Heidelberg, 2011.
- [22] L. H. P. Almeida, *McDSDS: a Multi-criteria Model for Planning Distributed Software Development Projects with Scrum*, [M.S. thesis], Graduate Program in Applied Informatics, University of Fortaleza, 2011.
- [23] M. Pedras, *Uma ferramenta de apoio ao gerenciamento de desenvolvimento de software distribuído*, UFPR, 2003.
- [24] E. H. Huzita, M. E. Pedras, G. Santiago, and T. F. Tait, "Dimanager: a tool for distributed software development management," in *International Conference on Enterprise Information Systems*, pp. 659–662, Porto, Portugal, 2004.
- [25] E. H. Huzita, T. F. Tait, and T. E. Colanzi, "Um ambiente de desenvolvimento distribuído de software-disen," in *I Brazilian Workshop on Distributed Software Development*, pp. 31–38, João Pessoa, Brazil, 2007.
- [26] E. Huzita, C. Silva, and I. Wiese, "Um conjunto de soluções para apoiar o desenvolvimento distribuído de software," in *II Brazilian Workshop on Distributed Software Development*, pp. 101–110, Campinas, Brazil, 2008.
- [27] S. Setamanit, W. Wakeland, and D. Raffo, "Planning and improving global software development process using simulation," in *Proceedings of the 2006 international workshop on Global software development for the practitioner - GSD '06*, pp. 8–14, Shanghai, China, 2006.
- [28] S. Setamanit, W. Wakeland, and D. Raffo, "Using simulation to evaluate global software development task allocation

- strategies,” *Software Process: Improvement and Practice*, vol. 12, no. 5, pp. 491–503, 2007.
- [29] S. Setamanit and W. Wakeland, “Improving global software development project performance using simulation,” in *PICMET '07 - 2007 Portland International Conference on Management of Engineering & Technology*, pp. 2458–2466, Portland, OR, USA, 2007.
- [30] P. Jalote and G. Jain, “Assigning tasks in a 24-h software development model,” *Journal of Systems and Software*, vol. 79, no. 7, pp. 904–911, 2006.
- [31] R. Prikladnicki, *MuNDDoS: um modelo de referência para desenvolvimento distribuído de software*, [M.S. thesis], PUCRS, 2003.
- [32] N. Mullick, M. Bass, Z. Houda, P. Paulish, and M. Cataldo, “Siemens global studio project: experiences adopting an integrated GSD infrastructure,” in *2006 IEEE International Conference on Global Software Engineering (ICGSE'06)*, pp. 203–212, Florianopolis, Brazil, 2006.
- [33] D. Mak and P. Kruchten, “Task coordination in an agile distributed software development environment,” in *2006 Canadian Conference on Electrical and Computer Engineering*, pp. 606–611, Ottawa, Canada, 2006.
- [34] A. Barcus and G. Montibeller, “Supporting the allocation of software development work in distributed teams with multi-criteria decision analysis,” *Omega*, vol. 36, no. 3, pp. 464–475, 2008.
- [35] V. S. dos Santos, T. A. B. Pereira, B. L. Ribeiro, and G. Elias, “Um framework de recomendação para alocação de equipes de desenvolvimento em projetos distribuídos de linhas de produto de software,” in *IV Brazilian Workshop on Distributed Software Development*, pp. 42–49, Salvador, Brazil, 2010.
- [36] S. U. Khan, M. Niazi, and R. Ahmad, “Factors influencing clients in the selection of offshore software outsourcing vendors: an exploratory study using a systematic literature review,” *Journal of Systems and Software*, vol. 84, no. 4, pp. 686–699, 2011.
- [37] S. U. Khan, M. Niazi, and R. Ahmad, “Critical success factors for offshore software development outsourcing vendors: a systematic literature review,” in *2009 Fourth IEEE International Conference on Global Software Engineering*, pp. 207–216, Limerick, Ireland, 2009.
- [38] S. U. Khan, M. Niazi, and R. Ahmad, “Barriers in the selection of offshore software development outsourcing vendors: an exploratory study using a systematic literature review,” *Information and Software Technology*, vol. 53, no. 7, pp. 693–706, 2011.
- [39] P. J. Ågerfalk, B. Fitzgerald, H. Holmström, B. Lings, B. Lundell, and E. Ó. Conchúir, “A framework for considering opportunities and threats in distributed software development,” in *Proceedings of the International Workshop on Distributed Software Development*, pp. 47–61, Paris, France, 2005.
- [40] J. Noll, S. Beecham, and I. Richardson, “Global software development and collaboration: barriers and solutions,” *ACM Inroads*, vol. 1, no. 3, p. 66, 2010.
- [41] M. Jiménez, M. Piattini, and A. Vizcaino, “Challenges and improvements in distributed software development: a systematic review,” *Advances in Software Engineering*, vol. 2009, Article ID 710971, 14 pages, 2009.
- [42] R. Kommeren and P. Parviainen, “Philips experiences in global distributed software development,” *Empirical Software Engineering*, vol. 12, no. 6, pp. 647–660, 2007.
- [43] A. B. Marques, J. R. Carvalho, R. Rodrigues, T. Conte, R. Prikladnicki, and S. Marczak, “An ontology for task allocation to teams in distributed software development,” in *2013 IEEE 8th International Conference on Global Software Engineering*, pp. 21–30, Bari, Italy, 2013.
- [44] R. Prikladnicki and J. L. N. Audy, “Process models in the practice of distributed software development: a systematic review of the literature,” *Information and Software Technology*, vol. 52, no. 8, pp. 779–791, 2010.
- [45] J. M. Verner, O. P. Brereton, B. A. Kitchenham, M. Turner, and M. Niazi, “Systematic literature reviews in global software development: a tertiary study,” in *16th International Conference on Evaluation & Assessment in Software Engineering (EASE 2012)*, pp. 2–11, Ciudad Real, Spain, 2012.
- [46] A. S. Khan and Z. Subhan, “Distributed software development process, initiatives and key factors : a systematic literature review,” *International Journal Of Multidisciplinary Sciences And Engineering*, vol. 5, no. 12, pp. 7–21, 2014.
- [47] M. Jiménez Monasor, *A Framework for Interaction Training in Global Software Development*, [Ph.D. thesis], University of Castilla-La Mancha, Ciudad Real, Spain, 2014.
- [48] M. Simão Filho, *An Approach Structured on Verbal Decision Analysis to Support the Allocation of Tasks in Projects of Distributed Software Development*, [Ph.D. thesis], University of Fortaleza, 2017.
- [49] M. Simão Filho, P. R. Pinheiro, and A. Bessa Albuquerque, “Applying verbal decision analysis in distributed software development: rank ordering the influencing factors in task allocation,” in *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1–6, Las Palmas, Spain, 2016.
- [50] M. Simão Filho, P. R. Pinheiro, and A. B. Albuquerque, “Task allocation in distributed software development aided by verbal decision analysis,” in *Software Engineering Perspectives and Application in Intelligent Systems*, pp. 127–137, Springer, Cham, 2016.
- [51] M. Simão Filho, P. R. Pinheiro, and A. B. Albuquerque, “Analysis of task allocation in distributed software development through a hybrid methodology of verbal decision analysis,” *Journal of Software: Evolution and Process*, vol. 29, no. 7, article e1867, 2017.
- [52] M. Simão Filho, P. R. Pinheiro, and A. B. Albuquerque, “Task assignment to distributed teams aided by a hybrid methodology of verbal decision analysis,” *IET Software*, vol. 11, no. 5, pp. 245–255, 2017.
- [53] M. Simão Filho, P. R. Pinheiro, and A. B. Albuquerque, “Towards a model for task allocation in distributed software development,” in *X Brazilian Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems*, pp. 1–10, Maringá, Brazil, 2016.
- [54] M. Simão Filho, P. R. Pinheiro, and A. B. Albuquerque, “Applying verbal decision analysis to task allocation in distributed development of software,” in *Proceedings of the 28th International Conference on Software Engineering and Knowledge Engineering*, pp. 402–407, Redwood City, San Francisco Bay, USA, 2016.



Submit your manuscripts at
www.hindawi.com