IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received April 11, 2019, accepted April 28, 2019, date of publication May 15, 2019, date of current version June 13, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2916782

Multilevel Ontology Framework for Improving Requirements Change Management in Global Software Development

ABEER ABDULAZIZ ALSANAD[1,2], AZEDDINE CHIKH[3], AND ABDULRAHMAN MIRZA[2]

[1]Information Systems/Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University, Riyadh 11432, Saudi Arabia
[2]Information Systems/Computer and Information Sciences, King Saud University, Riyadh 11362, Saudi Arabia
[3]Computer Sciences, University of Tlemcen, Tlemcen 13000, Algeria

Corresponding author: Abeer Abdulaziz Alsanad (aaasanad@imamu.edu.sa)

This work was supported by the Deanship of Scientific Research through the Initiative of DSR Graduate Students Research Support (GSR).

ABSTRACT Requirements engineering is one of the most important pillars of software engineering. Its success contributes greatly to that of the software as a whole. In fact, the software development process is not devoid from changing requirements, which affects the cost, time, and quality of the final software. The change problem is unavoidable and also swells when the development of the software is made globally. Therefore, there is a need to improve the quality of requirements change management (RCM), especially in global software development (GSD) environments. Our research hypothesis is that the RCM is naturally a knowledge-intensive process that can benefit substantially from ontology. Indeed, we assume that using a multilevel ontology framework will greatly support RCM in GSD environments by ensuring the semantic correctness of the requirement change request and accordingly solving miscommunication and misunderstanding problems. The framework was successfully evaluated using a questionnaire and a case study. The results indicate that using the proposed framework can intensely improve the semantic correctness of requirement change requests. Accordingly, the entire RCM process is then improved by increasing the reliability of the change and reducing the time consumed for dealing with semantically wrong change requests.

INDEX TERMS Requirement engineering, global software development, requirement change management, change requests, ontology.

I. INTRODUCTION

Software development is a thorny journey that faces many difficulties. Many of them hinder software quality that developers are very keen to address. In fact, software quality is the most important factor for software development and it directly related to the success of a software project [1]. Also, as software engineering continued its growth, its projects became more complex and need to be high quality software [2]. Also, the issues of communication and RCM in GSD require more efforts as compared to the centralized software development system [3]. One of the biggest threats to quality is the frequency of requirements change. Unfortunately, this change cannot be avoided [4]. This raises the need to improve its management and reduce its effects on the software development. The schedule, cost, and quality of software will be influenced [5]. Actually, managing requirements change efficiently, completely, and reliably is still a difficult problem for the software engineering community. Software reliability depends on time and cost, which are the two main factors affecting the quality of the software as a whole [6]. Indeed, change requesters write ambiguous and poorly justified change requests (CRs), on one hand. Requirements engineers face difficulties in objectively analyzing and implementing those requests, on the other hand. Therefore, many CRs are likely to be misunderstood, which leads to low quality of change management. In addition, although users' application domain knowledge in a software application domain plays an important role in writing CRs, not all users know how to objectively express understandable and non-risky change requests for requirements engineers. According to Holtkamp et al. [7], the most prominent challenges facing

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaobing Sun.

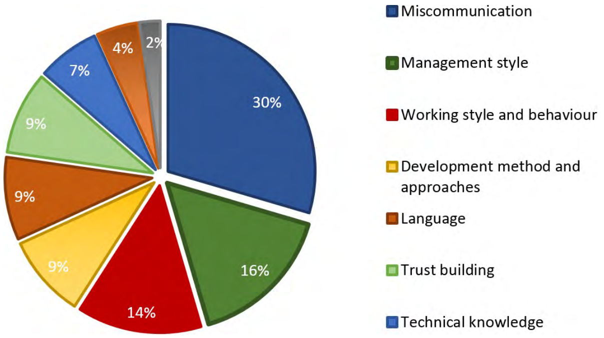

FIGURE 1. The challenges facing RCM in GSD, [7].

requirements change management (RCM) in global software development (GSD) environments are stated on Figure 1. In the figure, miscommunication problems represent the most RCM problems in GSD, making it the one most worth consideration.

Miscommunication in RCM during GSD leads to multiple risks [7], [8]. One considered in this paper is misunderstandings emerging from unclear or ambiguous requirements and CRs. Software development in GSD environments has recently become a source of interest. Most software organizations are directing towards GSD. This trend has increased lately due to the recognition of the enormous benefits of working in such environments [9]. Among GSD's benefits, expertise all around the world can be involved in the development process, costs can be reduced through using cheap labor, development occurs around the clock, and marketing will be much effective and faster [9]–[13]. However, the main challenge to overcome is miscommunication [14], [15]. This problem can be supported by enforcing the usage of a common language or an integrated set of related knowledge in CR [7]. From our point of view, the main reason for miscommunication and misunderstanding problems is a lack of explicit knowledge about three domains related to (1) RCM within GSD, (2) requirements engineering, and (3) the specific software application. We aim to solve these problems by improving the semantics of requirement CRs through using a common language. Therefore, we propose a multilevel ontology framework. Each level of the framework provides a specific kind of ontology. An ontology is a representational artifact used to define the semantics and meaning of the knowledge [16], [17]. In literature, researchers have created several ontology-based frameworks for RCM. However, most of them adopt one single domain ontology that does not cover all aspects related to RCM for GSD, such as software application, requirements engineering, and requirements change aspects. Hence, this paper, which goes beyond this research, claims that developing a multilevel ontology framework to support RCM process for GSD will ensure the correctness, completeness and logicality of the change. It gives a comprehensive set of ontologies to ensure all semantic aspects of the requirement CRs are met. In addition, it will allow domain users to easily express their CRs and allow the requirements engineers to objectively interpret and process them.

The remainder of this paper is structured as follows. Section 2 presents the related works. Section 3 presents the proposed multilevel ontology framework. The framework is evaluated in Section 4. The results and discussion are presented in section 5. Finally, section 6 concludes the current work and proposes some perspectives.

II. RELATED WORKS

A literature review shows few works addressing the problems of RCM in GSD using ontologies. Kumar S.A and Kumar T.A [11] studied the requirement management problem in GSD and stated its challenges. They designed a framework to address the problem. Moreover, they used an ontology-based knowledge-management system to solve some requirement problems, such as missing requirements and inconsistency found in the requirements; communication and knowledge management problems. Also, they used the framework to improve the management activities in the GSD environment. In addition, they proposed requirements management metrics with which to measure and manage software processes during the development of information systems. They stated that their framework effectively facilitates managing of requirement engineering. Lai and Ali [5] proposed an ontology-based requirements-management method for GSD. Their method consists of four stages: (1) establishing and maintaining a requirements repository that can be used among the team members, (2) generating a requirements traceability matrix to trace requirement through development process, (3) communicating and discussing requirements by preparing the agenda before meetings, and (4) requirements change management by creating graphs that make changes in requirements easier to understand. They used a case study for validation. Hafeez et al. [18] used an ontology to support the problem of RCM in GSD. However, their framework, which is defined at an abstract level, lacks validation from the software industry to evaluate its completeness and consistency. Khatoon et al. [19] addressed the problem of RCM in GSD by ensuring knowledge management and shared understanding through a software engineering ontology. They used a case study to evaluate their framework.

A summary of the four former works is displayed in Table 1. From the previous literature review, we noticed a lack of shared knowledge about the RCM process in GSD. Otherwise, using ontologies in this field was scarcely studied in the associated literature.

III. THE PROPOSED MULTILEVEL ONTOLOGY FRAMEWORK

In this paper, knowledge related to requirements change is considered the main component of the multilevel ontology framework. The term "framework" is defined in Macmillan Dictionary [20] as "a system of rules, laws, arguments etc. that establishes the way that something operates in business, polices, or society." The main purpose of designing

TABLE 1. A summary of related works using ontology in RCM for GSD.

Reference	Objective	Solution	Advantage	Limitation
Hafeez et al. [18]	To solve the problem of RCM in GSD	A framework for improving the efficiency of RCM in GSD environments	The framework provides a guideline to the developer team regarding how frequently changes could be incorporated.	Lacks validation from the software industry to evaluate its completeness and consistency. The framework is de- fined at in an abstract level.
Lai and Ali [5]	To solve the requirements management problem in GSD	An ontology-based requirements-management method for GSD	RCM will be easy and quicker.	They designed ontologies for a requirement repository, traceability, and establishing a communication between teams but did not design an RCM ontology.
Khatoon et al. [19]	Addressed the problem of RCM in GSD	Ontological technique	Assuring knowledge management, shared understanding and a common vocabulary through software engineering ontology.	Their ontology is general and needs more enhancements.
Kumar and Kumar [11]	Studied the requirement management problem in GSD and stated its challenges	Ontology-based knowledge-management systems	Helps to solve requirement problems such as missing or inconsistent requirements and communication and improve the management activities in the GSD environment	They improved requirement management in general but not RCM in particular.

the framework as a multilevel ontology is to guarantee the correctness of the requested change from three perspectives: (1) the requirements change, (2) the requirements engineering, and (3) the specific software application domain. The correctness of the semantic of change requests is ensured through a semantic authoring tool, based on the three ontologies. Our framework has the potential to improve two main factors of RCM: reliability and time. Therefore, our research hypothesis stated before can be split into two sub-hypotheses. We claim that our framework will (1) improve the RCM process through increasing the reliability of CRs (1^{st} sub-hypothesis) and (2) decrease the time needed for RCM process, especially in GSD environment (2^{nd} sub-hypothesis).

A. FRAMEWORK LEVELS

Our multilevel ontology framework is a system composed of three ontologies located at different levels. At each level, the corresponding ontology covers a specific perspective under which the CR semantics are checked. These domain ontologies cover (1) the domain of RCM in the GSD; (2) the domain of requirements engineering; and (3) the specific software application domain, such as human resources, sales, etc. Details about these three domain ontologies are presented in the following sections.

1) FIRST-LEVEL ONTOLOGY

The first level of the proposed framework corresponds to the requirement change ontology (RCO), which is dedicated to the domain of requirement change in GSD. A full description of RCO is presented in [21] and shown in Figure 2.

2) SECOND-LEVEL ONTOLOGY

The second level of the proposed framework corresponds to the ontology for the requirement engineering domain. A requirements' domain ontology specific to the software under development need to be chosen to represent this level.

3) THIRD-LEVEL ONTOLOGY

The third level of the proposed framework corresponds to the ontology for the software application domain. We can use an application domain ontology specific to the application of the software under development, such as a human resources ontology.

B. FRAMEWORK IMPLEMENTATION

In this section, implementation of multilevel ontology framework is described. The framework considered as an authoring module for requirements' CRs. Details of how the framework works is shown in Figure 3 and described in the following points:

- The input to the module is the framework's three ontologies. The RCO is used as the first level of the framework. Ontologies for the second and the third levels are provided (either adopted or built) in OWL (Web Ontology Language) based on the software to be developed.
- The three OWL files are converted to XML (Extensible Markup Language) files to ease the extraction of the ontologies' concept
- The three ontologies' concepts are extracted from the XML files and added to a database.
- A stop-words list, which is the common words in a language, are added to the same database.
- A requester can add additional common words. These words are added to the same database, also.
- The change requester could be any stakeholder from any of the software's development sites.
- A CR form is used as an editing tool to request a requirements change by a change requester. The change

FIGURE 2. RCO Protégé snapshot, [21].

description field in the form is linked to the framework's ontologies' concepts available in the database. Then, all the concepts of the three ontologies, stop words list and created words are shown to the change requester while editing the CR.

- The output from this module is a semantic CR.

This module partially ensures the correctness of the semantics of the authored CR.

A prototype for the multilevel ontology framework was implemented to show how the framework works. A change request form was designed to allow the change requester to fill the request. There are two main windows: Change Request Form and Add New General Terms windows, as shown in Figures 4 and 5. The change request form fields were designed following the change request class in RCO. All of the fields are either free text or drop-down lists except the change description field, which is the most important field since it will ensure the semantics of the CRs. In the change description section, text box has an auto-fill propriety to help the requester describe his/her change request, using concepts from the three framework's ontologies. Since the description will not be limited to ontologies' concepts, a general terms from stop words list were added to the list of auto-complete words. Moreover, since the requester may need to add a word that is not one of the stop words and not one of the three ontologies' concepts, a requester can add general terms through the general term window. Any word added here will be added to the auto-filled word list. The framework prototype uses the three ontologies' OWL files as an input. Then, the OWL files are converted to XML files to ease the extraction of the ontology concepts. The three XML files are searched to extract the concepts and then add them to the database. Also, stop words are added to the same database. When the requester fills the change request and reaches the change description, all of the concepts of the three ontologies are shown as auto-complete options from which to choose. Finally, after the change request is done, it can be printed or saved as a PDF file.

IV. MULTILEVEL ONTOLOGY FRAMEWORK EVALUATION

To decide on how to evaluate the proposed multilevel ontology framework, we looked at other research in the same field. Research methodologies used in research about RCM in GSD for 10 years from 2008 to 2017 are presented by Akbar et al. [22] and shown in Figure 6. The two most frequently used research methods are questionnaire surveys and case studies. Accordingly, we adopted these two methods to evaluate our framework and test our two hypotheses.

A. FRAMEWORK EVALUATION THROUGH A QUESTIONNAIRE SURVEY

For validating the proposed framework, a survey was conducted to obtain reliable opinions from a group of experts with a wide range of expertise about RCM. Expert opinions can be used to support a system, validate it, improve it, and increase its usability [23]. The aim of the survey is to validate the proposed framework and assess it against the two hypotheses based on reliability and time. A questionnaire was developed and conducted to assess the qualitative data. The questionnaire describes the idea behind the proposed framework. A number of RCM experts were asked to provide their opinions on the proposed framework. Their feedback was collected for analysis, and their responses and comments were taken into consideration.

1) QUESTIONNAIRE DESIGN

A Web-based survey was utilized to increase the ability to distribute it worldwide. The targeted populations for the survey were project managers, requirement engineers, change control board (CCB) members, change managers and key leaders associated with RCM in GSD. We used a list of email addresses generated from snowball sampling that was used by Shafiq et al. [24]. The list contained 111 email addresses. The list was doubled checked to ensure that the email addresses belong to experts in software engineering and RCM. In total, 46 experts were chosen and asked to participate on the survey.

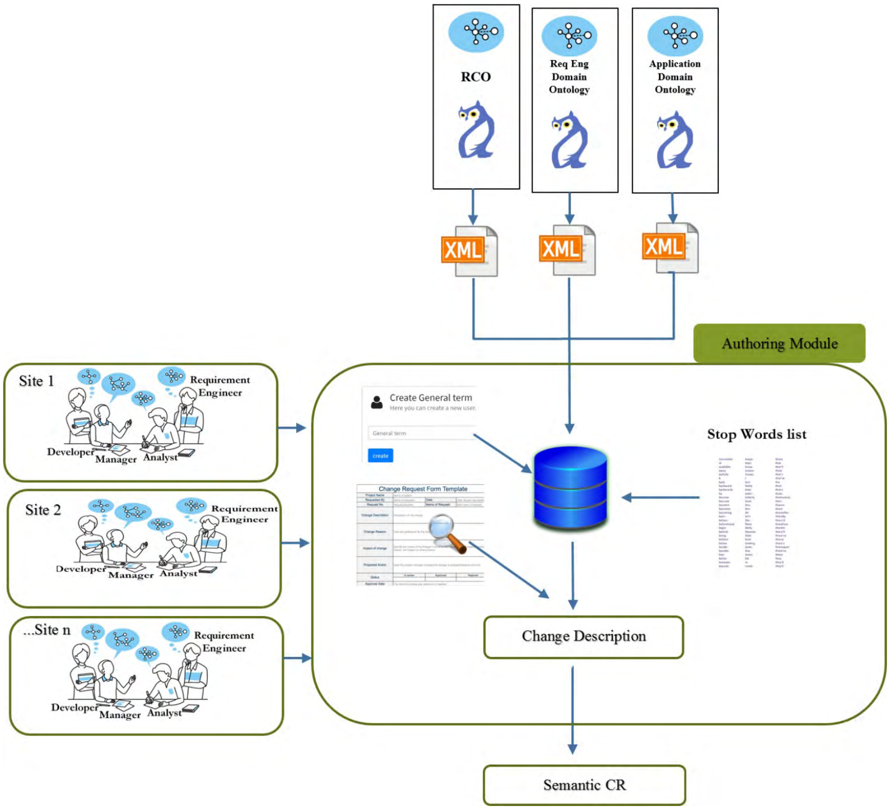

FIGURE 3. Multilevel ontology framework implementation.

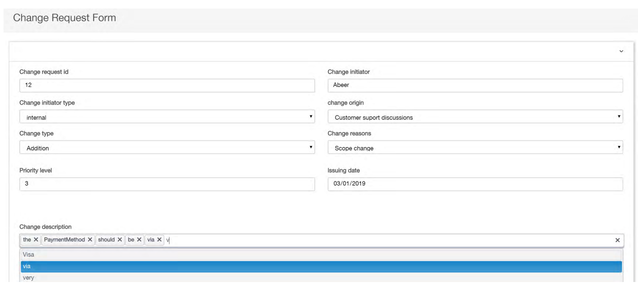

FIGURE 4. Change request form window snapshot.

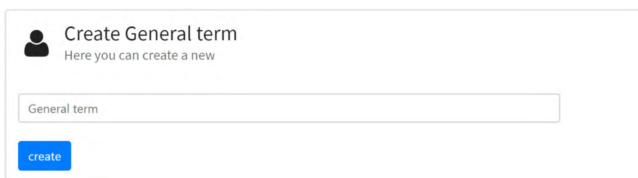

FIGURE 5. Create new general term window snapshot.

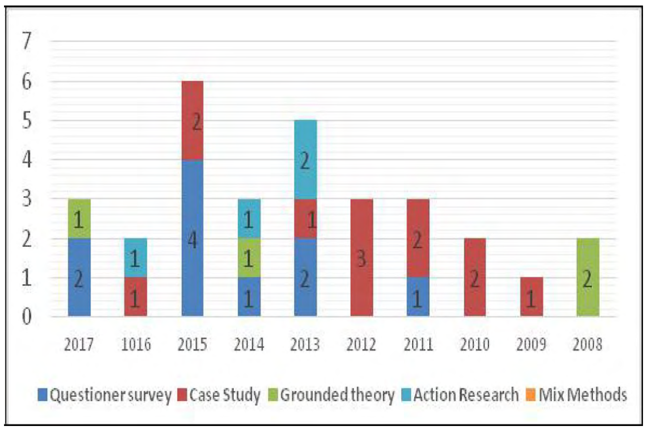

FIGURE 6. Analysis of used research methodologies with studies, [22].

The email addresses that could not be checked were removed from the list. Moreover, 33 other experts we found were asked to participate on the survey. They received participation requests through email or LinkedIn messages. Of the 79 experts, only 26 responded to the survey. The respondents were from different countries, as shown in Figure 7. The highest response rate was coming from KSA (Kingdom of Saudi Arabia), about 38.5%. The respondents' years of experience are shown in Figure 8. About 40% of them have an experience of 11 years and above.

2) QUESTIONNAIRE INSTRUMENT

Most of the survey questions evaluated ordinal information on a 5-point Likert scale ranging from "strongly agree" to "strongly disagree." The questionnaire components were collected on the basis of the research needs and then piloted for understanding, ambiguity, and fulfillment. The survey

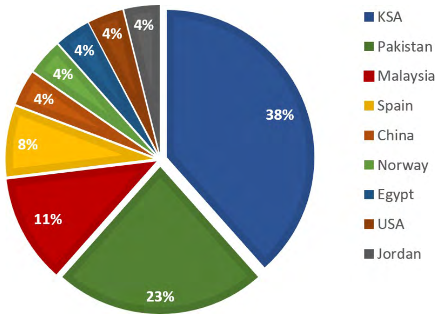

FIGURE 7. Respondents' place of residency distribution.

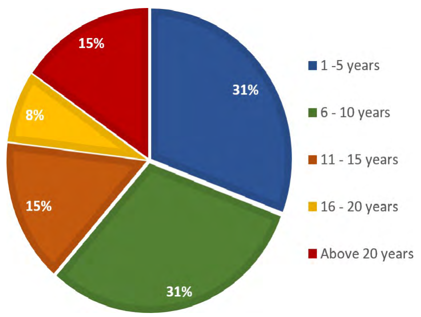

FIGURE 8. Respondents' length of expertise.

also included a cover letter stating the objectives and benefits of the survey and gives an overview of the proposed framework.

3) QUESTIONNAIRE PILOT STUDY

A pilot study was conducted after the questionnaire development with a sample of users to verify the clarity and structure of the questionnaire and its instructions. After that, all unclear statements were revised, and the respondents' comments on wording and structure were used to improve the questionnaire. After the pilot study, it was assumed that the questionnaire was ready to be completed by the participants.

4) QUESTIONNAIRE DIVISIONS

The conducted questionnaire consisted of 24 queries divided into five major sections: (1) six demographic questions; (2) seven problem significance questions; (3) four questions for reliability; (4) four questions for time; (5) and three open-ended questions regarding the framework in general.

B. FRAMEWORK EVALUATION THROUGH A CASE STUDY

The other method used to evaluate and test the proposed framework is the case study. Yin [25] defined a case study as ''an empirical inquiry that investigates a contemporary phenomenon within its real-life context'' (p. 13). The case study was chosen based on the following criteria:

1) The problem domain was understandable by the researchers as no terms were specific to the company.
2) The requirements were written as ontology based knowledge or can be built as ontology based knowledge by researchers.

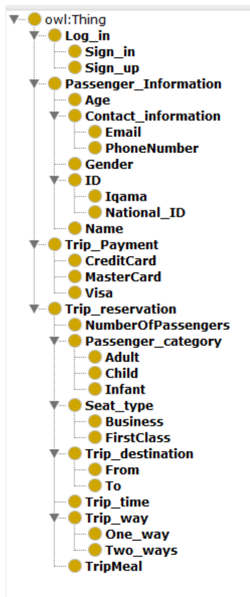

FIGURE 9. A snapshot of the RioMetro ontology.

3) The application domain has an existing ontology, or it can be built by the researchers.

Based on the criteria above, a case study was chosen for evaluating the proposed framework.

1) SELECTED CASE STUDY

The chosen case study was the RioMetro software application, which is dedicated to reserving a ticket to use the Riyadh Metro. The application domain is ticket reservations.

2) CASE STUDY IMPLEMENTATION

The case study was designed first by determining the framework's ontologies. The three ontologies composing the framework are RCO as a RCM ontology (first level), the RioMetro ontology as a requirement engineering domain ontology (second level), and the ticket ontology as a specific application domain ontology (third level). We built the RCO and the RioMetor ontologies. RCO was proposed in [21] while the RioMetor ontology was built for the purposes of the case study. A snapshot of the RioMetro ontology is shown in Figure 9. Tickets ontology is defined as ''a Web vocabulary for describing tickets for concerts, locations like museums, or any kind of transportation for e-commerce'' and found in [26]. The framework was applied to the case study as described in the framework implementation section.

V. RESULTS AND DISCUSSION

This section shows the obtained evaluation results, which are divided into two parts:

A. QUESTIONNAIRE SURVEY

In this section, the results of the questionnaire survey method are presented. In addition, the extent to which the

TABLE 2. Survey questions with respect to each factor.

Section	Question	Question #
Problem Significance	Have you ever encountered a problem of receiving a request for a requirement change written semantically wrong.	Q2.2
	Writing a semantically wrong change request is a critical problem needs handling.	Q2.3
	The possibility of having semantic errors in change requests rises more in GSD environment.	Q2.4
	The main reason of writing semantically wrong change requests in GSD is miscommunication problem.	Q2.5
	Semantic correctness of the change request can be ensured by using ontologies for the software domain.	Q2.6
	Using the proposed framework will effectively ensure the correctness of the semantic of the change requests.	Q2.7
Reliability	Semantic errors found in requirement change request reduce its reliability.	Q3.1
	Reducing the reliability of change requested affects negatively the quality of the RCM process.	Q3.2
	Using the proposed framework will increase the reliability of change request.	Q3.3
	Increasing the reliability of the change request increases the quality of the RCM process.	Q3.4
Time	Semantic errors found in requirement change requests consumes additional time in understanding and correcting these requests and even worse it can lead to applying a wrong change request which will waste even more time.	Q4.1
	Consuming additional time (time wasting) in handling semantically wrong change requests affects negatively the quality of RCM process.	Q4.2
	Using the proposed framework will leads to avoid wasting time in handling semantically wrong change requests.	Q4.3
	Reducing the time wasted due to wrong change requests increases the quality of the RCM process.	Q4.4

TABLE 3. Summary of survey responses.

Section	Question	SD	D	N	A	SA	Total
Problem Significance	Q2.2	0%	26.9%	38.5%	30.8%	3.8%	100%
		0	7	10	8	1	26
	Q2.3	0%	3.8%	7.7%	57.7%	30.8%	100%
		0	1	2	15	8	26
	Q2.4	0%	3.8%	38.5%	46.2%	11.5%	100%
		0	1	10	12	3	26
	Q2.5	0%	11.5%	19.2%	53.8%	15.4%	100%
		0	3	5	14	4	26
	Q2.6	0%	0%	34.6%	61.5%	3.9%	100%
		0	0	9	16	1	26
	Q2.7	0%	12%	12%	56%	20%	100%
		0	3	3	14	5	25
Reliability	Q3.1	0%	0%	8%	72%	20%	100%
		0	0	2	18	5	25
	Q3.2	0%	3.9%	11.5%	61.5%	23.1%	100%
		0	1	3	16	6	26
	Q3.3	0%	0%	34.6%	42.3%	23.1%	100%
		0	0	9	11	6	26
	Q3.4	0%	0%	11.5%	57.7%	30.8%	100%
		0	0	3	15	8	26
Time	Q4.1	0%	3.8%	15.4%	46.2%	34.6%	100%
		0	1	4	12	9	26
	Q4.2	0%	0%	15.4%	53.8%	30.8%	100%
		0	0	4	14	8	26
	Q4.3	0%	3.8%	23.1%	57.7%	15.4%	100%
		0	1	6	15	4	26
	Q4.4	0%	7.7%	23.1%	57.7%	11.5%	100%
		0	2	6	15	3	26

framework achieves the desired objectives is also analyzed. The quantitative results of the questionnaire are reported in graphs and tables. The respondents' demographic information shows an adequate representation of the required population. Table 2 shows survey questions for only the second, third, and fourth sections, which were dedicated to assessing the problem's significance and also the reliability and time factors while Table 3 summarizes the questionnaire responses for these sections. The open-ended questions include three inquiries. The first is about the respondents' opinions on the framework as a whole. Most of their opinions are positive. They think that the framework is useful, interesting, well-conceived, easy to understand, and easy to follow; further, they believe that it has a significant positive impact on

TABLE 4. Numerical responses corresponding to each Likert scale response.

Response to Likert scale	Response score
Strongly disagree	5
Agree	4
Neutral	3
Disagree	2
Strongly disagree	1

TABLE 5. Mean analysis.

Factor	Mean
Problem significance	3.70
Reliability	4.06
Time	3.96

GSD organizations. The second question pertains to other improvements—rather than reliability and time—that the respondents think the framework can support. Most respondents feel that the framework can support cost efficiency and usability. The third question assesses what improvements the respondents can suggest for the framework. Suggestions was made to simplify the ontological form initiation so that it is less abstract for quick creation of change requests as we live in a rapid changing in economy and business and increasing competition challenge.

Two Likert scales were used; one is shown in Table 4. The calculated means based on this scale are shown in Table 5. The mean results indicate that the proposed framework performed effectively in terms of problem significance, reliability and time factors. However, the reliability factor dominated.

The other scale categorized the questionnaire's second, third, and fourth sections into three categories instead of five: a) strongly agree + agree, b) neutral, and c) strongly disagree + disagree. For presenting the results, only part A will be used, which shows the positive respondents' opinions about each factor of the frameworks. The questions with always, usually, often, rarely and never choices, were categorized into two categories instead of five. a) always + usually + often, b) rarely + never. For presenting the results, only part A will be used. The results are presented in Figure 10. From the figure,

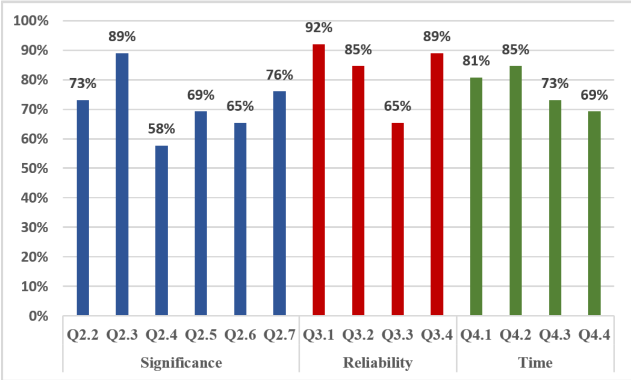

FIGURE 10. The percentage analysis for the significance of the problem, reliability factor and time factor.

the studied problem clearly has high significance since the percentage ranged of all respondents who agreed or strongly agreed on the significance of the problem started from 58% and reached 89%. Moreover, the reliability factor indicated a high response since most of the respondents agreed or strongly agreed on the importance of the reliability factor and the effectiveness of the framework in supporting this factor. The percentage ranged between 65% and 92%. Moreover, the time factor indicated a high response since most of the respondents agree or strongly agree on the importance of the time factor and the effectiveness of the framework in saving time. The percentage ranged between 69% and 85%.

The summary of the questionnaire result is as follows:

1) The problem of having semantically wrong change requests is a significant problem that needs handling.
2) The reliability factor is an important factor that affects the RCM process, and the framework will greatly ensure the reliability of CRs.
3) The time factor is an important factor that affects the RCM process, and the framework will greatly reduce the time wasted to understand or handle semantically wrong CRs.

B. CASE STUDY

The case study was applied on the framework prototype and it was successful. All of the concepts of the three ontologies were shown for the change requester to choose from. Also, general terms available in the stop words list, were presented. Change requesters using the same ontological concepts will guarantee the correctness of the CR and increases its reliability because it will be based on the three levels of ontologies, which cover all possible concepts in the domain of development. Accordingly, the time spent handling semantically wrong or ambiguous CRs will be reduced.

VI. CONCLUSION AND FUTURE WORK

In conclusion, a multilevel ontology framework was pro-Problem Significance posed. The aim of the framework is to improve the quality of RCM in GSD. This improvement was assessed through two factors: increasing the reliability of the CRs and decreasing the time spent on the RCM process through decreasing semantic errors in CRs. The results of the questionnaire and the case study showed the effectiveness of the framework and proved the paper's hypothesis. The multilevel ontology framework provides many advantages: it implements requirements changes more accurately and inexpensively by increasing human comprehension and understanding of CRs, implements change much faster than before, and semantically validates CRs by linking them to concepts from the three ontologies related to those CRs. To conclude, the proposed multilevel ontology framework was successfully evaluated using a combination of surveys and a case study and the research hypothesis along with its two parts were supported.

In the future, we suggest adding a querying module to the framework. The main purpose of this module will be to allow any stakeholder to query the ontologies. The module takes the three ontologies as inputs and outputs the result of the requested query. Moreover, we suggest, for the purposes of expanding the framework and increase its efficiency, to make the second level (the requirements engineering domain ontology) concerned with not only one ontology but multiple ontologies, thus providing different perspectives.

REFERENCES

[1] M. A. Akbar, J. Sang, A. A. Khan, F.-E. Amin, Nasrullah, M. Shafiq, S. Hussain, H. Hu, M. Elahi, and H. Xiang, "Improving the quality of software development process by introducing a new methodology—AZ-model," *IEEE Access*, vol. 6, pp. 4811–4823, 2018.

[2] M. A. Akbar, J. Sang, A. A. Khan, F.-E. Amin, Nasrullah, S. Hussain, M. K. Sohail, H. Xiang, and B. Cai, "Statistical analysis of the effects of heavyweight and lightweight methodologies on the six-pointed star model," *IEEE Access*, vol. 6, pp. 8066–8079, 2018.

[3] M. A. Akbar, Nasrullah, M. Shafiq, J. Ahmad, M. Mateen, and M. T. Riaz, "AZ-model of software requirements change management in global software development," in *Proc. Int. Conf. Comput., Electron. Elect. Eng. (ICE Cube)*, Nov. 2018, pp. 1–6.

[4] S. Jayatilleke and R. Lai, "A systematic review of requirements change management," *Inf. Softw. Technol.*, vol. 93, pp. 163–185, Jan. 2018.

[5] N. Ali and R. Lai, "A method of requirements change management for global software development," *Inf. Softw. Technol.*, vol. 70, pp. 49–67, Feb. 2016.

[6] A. Mateen and M. A. Akbar, "Estimating software reliability in maintenance phase through ann and statistics," May 2016, *arXiv:1605.00774*. [Online]. Available: https://arxiv.org/abs/1605.00774

[7] P. Holtkamp, I. Lau, and J. M. Pawlowski, "How software development competences change in global settings—An explorative study," *J. Softw., Evol. Process*, vol. 27, pp. 50–72, Jan. 2015.

[8] A. A. Khan, S. Basri, and P. D. D. Dominic, "Communication risks in GSD during RCM: Results from SLR," in *Proc. Int. Conf. Comput. Inf. Sci. (ICCOINS)*, Jun. 2014, pp. 1–6.

[9] S. U. Khan, M. Niazi, and R. Ahmad, "Critical barriers for offshore software development outsourcing vendors: A systematic literature review," in *Proc. 16th IEEE Asia–Pacific Softw. Eng. Conf.*, Penang, Malaysia, Dec. 2009, pp. 79–86.

[10] B. Javed and S. S. Minhas, "Process support for requirements engineering activities in global software development: A literature based evaluation," in *Proc. Int. Conf. Comput. Intell. Softw. Eng. (CISE)*, Wuhan, China, Dec. 2010, pp. 1–6.

[11] S. A. Kumar and T. A. Kumar, "Study the impact of requirements management characteristics in global software development projects: An ontology based approach," *Int. J. Softw. Eng. Appl.*, vol. 2, no. 4, p. 107, 2011.

[12] G. N. Aranda, A. Vizcaíno, and M. Piattini, "Analyzing and evaluating the main factors that challenge global software development," *Open Softw. Eng. J.*, vol. 4, no. 1, pp. 14–25, 2010.

[13] H. H. Khan, M. Mahrin, and B. Chuprat, "Factors generating risks during requirement engineering process in global software development environment," *Int. J. Digit. Inf. Wireless Commun.*, vol. 4, no. 1, pp. 63–78, 2014.
[14] P. Parviainen and M. Tihinen, "Knowledge-related challenges and solutions in GSD," *Expert Syst.*, vol. 31, no. 3, pp. 253–266, 2014.
[15] A. A. Khan, S. Basri, P. D. D. Dominic, and F. E. Amin, "Communication risks and best practices in global software development during requirements change management: A systematic literature review protocol," *Res. J. Appl. Sci., Eng. Technol.*, vol. 6, no. 19, pp. 3514–3519, 2013.
[16] R. E. Jensen, "Why closely coupled work matters in global software development," in *Proc. 18th Int. Conf. Supporting Group Work*, Nov. 2014, pp. 24–34.
[17] D. Allemang and J. Hendler, *Semantic Web for the Working Ontologist: Effective Modeling in RDFS and OWL*. Amsterdam, The Netherlands: Elsevier, 2006.
[18] Y. Hafeez, M. Riaz, S. Asghar, H. Naz, S. M. M. Gilani, A. Batool, M. Ahmed, and M. S. Hassan, "A requirement change management framework for distributed software environment," in *Proc. 7th Int. Conf. Comput. Converg. Technol. (ICCCT)*, Dec. 2012, pp. 944–948.
[19] A. Khatoon, Y. Hafeez, and T. Ali, "An ontological framework for requirement change management in distributed environment," *Nucleus*, vol. 51, no. 2, pp. 291–301, 2014.
[20] (2018). *Macmillan Dictionary*. [Online]. Available: http://www.macmillandictionary.com/dictionary/british/framework
[21] A. A. AlSanad, A. Chikh, and A. Mirza, "A domain ontology for software requirements change management in global software development environment," *IEEE Access*, vol. 7, pp. 49352–49361, 2019.
[22] M. A. Akbar, Nasrullah, M. Shameem, J. Ahmad, A. Maqbool, and K. Abbas, "Investigation of project administration related challenging factors of requirements change management in global software development: A systematic literature review," in *Proc. Int. Conf. Comput., Electron. Elect. Eng.*, Quetta, Pakistan, Nov. 2018, pp. 1–7.
[23] C. Okoli and S. D. Pawlowski, "The Delphi method as a research tool: An example, design considerations and applications," *Inf. Manage.*, vol. 42, no. 1, pp. 15–29, 2004.
[24] M. Shafiq, Q. Zhang, M. A. Akbar, A. A. Khan, S. Hussain, F.-E. Amin, A. Khan, and A. A. Soofi, "Effect of project management in requirements engineering and requirements change management processes for global software development," *IEEE Access*, vol. 6, pp. 25747–25763, 2018.
[25] R. Yin, *Case Study Research, Design and Method*, T. Oaks, Ed. Newbury Park, CA, USA: Sage, 2003.
[26] M. Hepp. (2010). *Tickets Ontology*. [Online]. Available: http://www. hepp-netz.de/ontologies/tio/ns

ABEER ABDULAZIZ ALSANAD received the bachelor's degree in computer science from Prince Sultan University and the master's degree in information systems from King Saud University, where she is currently pursuing the Ph.D. degree with the Information Systems Department, College of the Computer and Information Sciences. She is also a Lecturer with the Information Systems Department, College of Computer and the Information Sciences, Imam Mohammad Ibn Saud Islamic University. She has published several conference and journal papers. Her major interests are software engineering, requirement engineering, and change management.

AZEDDINE CHIKH received the Ph.D. degree in information systems from the National Institute of Computer Sciences, Algeria and Paul Sabatier University, France, in 2004, the master's degree in information systems from the National Institute of Computer Sciences, Algeria, in 1994, the master's degree in e-Learning from Switzerland, in 2007, and the Graduate Certificate in systems engineering from the University of Missouri, Rolla, MO, USA, in 2003. He is currently a Professor of computer sciences and the Director of the Research Laboratory LRIT, University of Tlemcen, Algeria. He has worked previously as an Associate Professor with King Saud University, Saudi Arabia. His research interests include learning design, communities of practice, the semantic web, and software requirements engineering.

ABDULRAHMAN MIRZA received the Ph.D. degree in computer science with the Illinois Institute of Technology. He is a Professor of information systems with the King Saud University (KSU). He is currently a Consultant with the Deputyship of Planning and Development, Ministry of Education, and the acting Director of the Center for Research on Educational Policies. Some of his previous leadership positions include Vice Dean of Academic Affairs at the College of Computer and Information Sciences, KSU, and the General Supervisor of the General Directorate of Teachers Affairs at the Ministry of Education. He has also served as a Senior Advisor to the Minister of Education, and to the Minister of Higher Education. He also held other positions, such as the Director of Quality and Accreditation with the Saudi Electronic University, the Deputy Director of the Center of Excellence in Information Assurance, the Chairman of the Information Systems Department, and the CIO at the King Abdullah Foundation for Developmental Housing. His research interests include software engineering, e-commerce, and information security.