Information Technology and Management
https://doi.org/10.1007/s10799-021-00343-7

Towards aspect based requirements mining for trace retrieval of component-based software management process in globally distributed environment

Sadia Ali[1] · Yaser Hafeez[1] · Mamoona Humayun[2] · N. Z. Jhanjhi[3] · Dac-Nhuong Le[4]

Accepted: 29 September 2021

Abstract

Stakeholder satisfaction is a significant aspect of component-based product development. Satisfaction level of stakeholder varies due to diverse reviews and perspective about components functionalities. The reviews and perspective create ambiguities and misunderstanding during management of components requirement from specification to linking requirements that lead to product failures. The improper components management increases efforts and errors when component's stakeholders and development team is located in a globally distributed environment. The major issues of distributed component-based systems, are control, communication, coordination, and semantical analysis of different reviews and perspectives. As requirements of components is elicited and developed at different locations which created ambiguities and irrelevancy during components integration. Therefore, in this study, we proposed a framework to improve the management process of components requirement in a distributed environment. To reduce ambiguities and incompleteness among requirements, aspect based sentiment analysis has been utilized for each stakeholders' reviews and perceptive individually. On the other hand, to reduce involvement of stakeholder and efforts in components prioritization and linking processes, we adapted cased based reasoning method and decision tree-based classification of requirements, respectively. The performance of the proposed framework has been evaluated through an experimental approach in order to compare it with current practices i.e. Random selection and expert based. The findings described that the accuracy of component management in global development increases with proposed framework. Further, results show that there is an increase in product quality with decrease in irrelevancy and redundancy in stakeholders' aspects and priority.

Keywords Software management process · Component-based systems · Components management · Specification · Semantic analysis · Traceability · Aspect extractions · Case-based reasoning

1 Introduction

Software engineering now a days consider component-based systems (CBS) for complex and large systems development. The CBS application divide requirements into different components and provides different customization facility to fulfil massive stakeholders' needs and help them with reusability properties of CBS [3, 30, 36, 37]. For CBS development stakeholders' reviews and satisfaction are important without failures and less maintenance

✉ Dac-Nhuong Le
nhuongld@dhhp.edu.vn

Sadia Ali
sadiaalief@gmail.com

Yaser Hafeez
yasir@uaar.edu.pk

Mamoona Humayun
mahumayun@ju.edu.sa

N. Z. Jhanjhi
noorzaman.jhanjhi@taylors.edu.my

1 University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University, Rawalpindi, Pakistan

2 Department of Information systems, College of Computer and Information Sciences, Jouf University, Sakakah, Saudi Arabia

3 School of Computer Science and Engineering, Taylors University, Subang Jaya, Malaysia

4 Faculty of Information Technology, Haiphong University, Haiphong, Vietnam

Published online: 05 November 2021

costs. As product requirements are gathered and prioritized from stakeholder by considering their homogenous perspectives. At the point of satisfaction reviews and analysis, analyst judge distinctive point of views in the form of negative, positive or neutral due to diverse perspective and use of product [6, 8, 27, 56]. Therefore, reviews and satisfaction level are the most important factors to reduce product failures, high maintenance cost and issues in version development in continuous evolution environment [6, 27, 37, 38, 56]. Thus, stakeholders play vital role in CBS product development and in modern era, to get advantages of advance technology, online communication and coordination increases among organization.

Nevertheless, stakeholders and organization in online communication and coordination environment are based on distributed location for global software development (GSD). Thus, product development involves communication, teamwork, control and coordination [27, 47]. GSD placed all development teams at one place using different online web applications and communication tools to get advantages of advance technology and experts' skills worldwide [4, 12, 20, 31]. GSD environment development team and stakeholders are from different areas of world, this creates coordination, control and knowledge management issues [19, 39, 48]. These issues badly affect every phase of development from requirement specification to requirement maintenance due to ambiguous requirements, redundancy and irrelevancy in priority [6, 38]. These issues can be improved by semantical analysis of requirements, accurate priority based on diverse perspectives and correct linking of requirements [13, 19].

In the development of product, the main and challenging phase is component management (CM) during development of software for high product quality and stakeholder's satisfaction [6, 27, 53]. The CM consists of requirement analysis, specification, prioritization and traceability [6, 25, 35, 57]. In analysis phase, requirements based on reviews, user experiences and requirements documents are analyzed. Specification based on analysis phase to extract requirements from documents which written in natural language and by considering all stakeholders perspective homogeneous instead of semantics and aspect analysis of requirements documents. If analysis and specification are performed correctly, then accurate priority is assigned for correct implementation. Then accurate trace link is created for proper management of requirements and handling modification in requirements. However, CM phase creates ambiguities, incompleteness and inaccuracy more in GSD due to diverse perspectives of stakeholders and coordination challenges. To improve user experience and higher product, there is a need to analyze stakeholders' perspectives semantically along with accurate priority and correct linking of requirements [3, 8, 36, 45] However, most of products now a days are developed for massive customers' requirements satisfaction and consists of product different version and series instead of single and small product in dynamic development [35].

Therefore, diverse practices implemented for CM according to stakeholder's cost, nature and time of the project [1, 8, 17, 23]. Existing research is focusing on enhancing user experience and satisfaction semantically with sentiment analysis. Therefore, to reduce ambiguities, incompleteness and irrelevancy in semantic analysis for requirements, aspects-based sentiment analysis (ABSA) is used. ABSA is normally used to extract useful information to remove natural language drawbacks and different aspects in analysis like cost, services, environment etc. along with positive, negative or neutral impact of product to improve quality of product [20, 38, 56]. To improve the quality of product, requirements are categorized based on sentiment classification by reading all sentences and documents using aspect-based sentiment analysis (ABSA) [20, 35].

After extraction of requirements from different aspects, the main issue is to identify correct priority of aspects implementation. The most of aspects priority may be ambiguous during prioritization process [9, 10, 26, 27]. As for prioritization process, priority of requirements is important from diverse perspective of stakeholders. Thus, priority of every system is important for accurate implementation of requirements to reduce irrelevancy and redundancy in version based or family line products where most of requirements are reused to reduce complexity [5, 8, 17, 36]. Subsequently, to describe requirement type critical to rank for stakeholders. The scalability issue not resolved in large requirements set to avoid error, complexity and more resources consumption by most of prioritization practices [2, 8, 9, 32, 37]. Therefore, for extracting previous priority reuse for similar cases, case-based reasoning (CBR) technique is used. The CBR method of artificial intelligent (AI) techniques [49, 50, 55] and match each query to reuse with updated decision for prioritizing requirements of product.

After specification and prioritization process there is a need to link or identify relationship among all requirements aspects to verify and validate them during all phases of development. Therefore, for accurate trace links among all aspects from aspects extraction to increase accuracy, decision tree classifier is used. The requirement traceability is a practices to link requirements life during project development from their source to maintenance [2, 12, 22, 25]. It reduce complexity and help in error detection during development process to guarantee that relevant requirements are implemented and the source code is consistent with its requirement [8, 12, 25]. After ABSA for prediction of missing values based on association rules, features are classified to increase efficiency and accuracy like J48 machine learning algorithm's significant performance [12].

Therefore, to overcome the problems of requirement management process in GSD environment like ambiguity, coordination and control mechanism [8, 9, 12, 25, 58], a framework has been proposed to improve product quality in GSD within limited resources.

Research contributions

The research work consists of following contributions:

1. A framework has been proposed for the semantic based component requirement management from specification to mapping/ linking requirements with increase in diverse perspective stakeholder satisfaction level and software quality.
2. For components specification, ABSA technique has been utilized to analyze the components semantically, then for query-based association analysis used CBR technique and map priorities to get prioritized components requirement list.
3. For higher accuracy and performance automatic machine learning prediction technique has been used and for improving communication, coordination and control issue, team foundation server repository has been utilized.
4. For performance evaluation of proposed framework, experiment has been conducted to identify improvement in requirements management process during product development.
5. The results of study provide roadmap for practitioners and researchers for improving requirement management process in distributed industry.

The remaining study is designed as; Sect. 2 consist of related work to formulate motivational statement according to existing problems in current practices. Framework is proposed in Sect. 3, to provide solutions for described problem. Subsequently, results and discussion is explained in Sect. 4, to further elaborate steps and finding of experiment performed for evaluation. Conclusion and future work is described to summarize research study along with future dimensions in Sect. 5 respectively.

2 Related work

Specification and prioritization issues during CM highlight in different existing studies. Therefore, some studies highlighted issues in specification process and some in accurate prioritization or in generation of correct trace links. In software engineering all of these aspects are equally important in CM process and interlinked with each other during development process. Therefore, various CM approaches are required to be compared and checked with respect to different factors from existing studies to provide comprehensive solution instead of different solution. As requirements are required to verify and validate at every stage of development for higher satisfaction and good quality.

Thus, for requirements, assurance pursues to improve and maximize the chances of requirements quality through analysis and evaluation. The authors [44] present a process that used statistical and text-mining procedure to increase traceability assurance and minimize effort. Therefore, uses both requirements dissimilarity and similarity. Requirements prioritizing (RP) focus on stakeholders' feedback and brings a noteworthy cost and time due to maximum stakeholders' interactions. The study presented framework [7] to identify correct stakeholders' priority with less interactions in semi-automatic way. Similarly, Perini et al. [42] presented case-based ranking procedure for prioritization to facilitate requirements of stakeholder's. According to [43], presented Search-Based procedure for requirements selection and prioritization by analyzing and classifying issues in system version release. Hence, authors [52] presented Model based on neural network and fuzzy analytic hierarchy process for selection of appropriate stakeholders to improve their priority and satisfaction level.

The correct and accurate dependency, relationship and mapping among different artifacts during development of system identified through traceability. Therefore, different traceability performed in existing literature to link artifacts are; forward (link artifacts from requirements extraction to implantation or end information), backward (link artifacts from requirements to source of requirements or initial information), horizontal, vertical tracing, pre and post requirement traceability [29, 54, 57].

The CM (i.e. specification, prioritization and traceability) process complex in distributed environment due to lack of collaboration among teams and stakeholders. GSD widely spread in developing organization to utilize advance and multiple resource from all over the world. The developing in GSD environment distributed software development work according to their available resources and expertise. On accord to [33], in GSD organization requirements management is the important step during development and highlighted important issues of CM in GSD i.e. coordination, communication, specification, analysis, prioritization, team and stakeholders collaboration, control, and traceability.

To resolve these issues in GSD, existing studies in literature presented various procedures and practices but still there is scope of improvement in RM process during GSD software development effectively. Therefore, Dekhtyar and Hayes [21] presented technique for improvement of CM process in GSD environment. The study defined that traceability is the main solution to resolve CM and equally important during implementation and verification of software requirements. As traceability improve quality and satisfaction level along with reduction in testing efforts and cost

[18]. Subsequently, some other existing studies [2, 8–10, 16, 17, 21, 27, 36, 40, 46, 57] highlighted requirements specification, prioritization and traceability challenges should resolved for the improvement of software quality, stakeholders high satisfaction level and reducing maintenance cost.

The existing practices for CM have different challenges which required enhance technique for CM according to review and analysis of existing relevant studies identified from literature search. The enhancement based on mitigation of these challenges which we extracted during review and analysis such as no comprehensive process for overall CM improvement, specification incorrect and ambiguous, incomplete and incorrect combination of requirements due to improper priority issues, stakeholders involvement more which causes frequent changes, lack of improper expertise for requirement analysis after collection, no mapping of requirements from its origin to implementations, scalability issues, and complex to implement and verify new or change requirement. The literature comparative analysis of identified challenges described and compared in Table 1. For comparison of challenges we highlight them with terms i.e. specified (defined in relevant study), partially specified (defined similar challenges) and not specified (not defined in relevant study) in the literature.

The literature highlighted challenges for management of components requirements in GSD and describes outcomes of literature provide a guideline to researchers for developing theory about challenges of semantic based components management from specification to traces retrieval. Therefore, important issues identified from literature for CM in GSD are required to improve CBS development process are [2, 4, 6, 8, 17, 25, 36, 51];

– Coordination, communication and coordination among stakeholders and project team: As both stakeholders and project teams located distributed and components mostly

Table 1 Comparative analysis of literature

Parameters	(Lahon, et al. 2016)	(Guo et al. 2017)	(Alsanad et al. 2019)	(Ayala et al. 2018)	(Arias et al. 2018)	(Ali et al. 2019)	(Borg et al. 2019)	(Chatzipetrou et al. 2019)	(Kamal et al. 2020)	(Lu et al. 2020)
Semantic Analysis	■□	■■	■■	■■	■■	■■	□□	□□	■□	■■
Irrelevancy	■□	■□	■□	■□	■□	■■	■■	■■	□□	■■
Redundancy	■□	■□	■□	■□	■□	■■	■■	□□	□□	■□
Component Management	■□	■□	■■	■■	■■	■■	■■	■□	□□	■□
Communication	□□	□□	■■	□□	□□	■□	■□	■□	■■	□□
Coordination	□□	□□	■■	■□	□□	■□	■□	□□	■■	□□
Control	□□	□□	■■	■□	■□	■□	■□	■■	■■	□□
Changing Components Requirements	■□	□□	■■	■■	■□	■■	■■	■□	■■	■□
Multi perspective	□□	■□	■□	■□	■□	■■	□□	□□	■□	■□
Term Mismatch	■□	■■	■■	■■	■■	■■	■□	□□	□□	□□
Ambiguous Trace Links	□□	■■	□□	□□	■■	■■	□□	□□	□□	□□

Specified = ■■ Partially Specified = ■□ Not Specified = □□

developed by third party. Therefore, syncing the activities and components reusability created ambiguities and incompleteness.

- *Multi perspective:* The stakeholders of CBS applications have diverse perspective and there is different customization options available to stakeholders. The multi perspective of stakeholders not recognized semantically.
- *Semantic analysis*: If multi perspective analyze without semantic analysis then there is ambiguities and inconsistency among CBS customized options and components development.
- *Components prioritization and selection*: The selection and prioritization process during CM in GSD impacted due to incompleteness, ambiguities and without semantic analysis in requirements of components and less coordination with stakeholders.
- *Irrelevancy and redundancy*: Due to improper components requirements specification and analysis created irrelevancy and redundancy during components integrations.
- *Trace links*: Then due to above problems components correct relationship not identified which may failed the system.
- *Validation of components*: at the end, due to all above problem components validations process failed and increases faults with reduction in faults identification rate.

Therefore, to resolved all above challenges we proposed comprehensive framework and described in next section. The framework improved CM process and mapped requirement from its collection to selection.

3 Methodology

In the following section, presented and described proposed framework procedure for resolving problems identified from existing studies during review of literature. For identification of problems, development of proposed framework and evaluation process we adopted different steps as depicted in Fig. 1.

3.1 Proposed framework

In proposed framework for CBS component requirement management in globally distributed environment (CRMGDE) solution has been provided for mitigation of specification, prioritization and traceability problems during product development. Therefore, CRMGDE explained in Fig. 2.

3.1.1 Specification

In specification steps requirements extracted and analysis semantically based on different stakeholders' perspectives using reviews and perspective as input. The input has been used to formally describe requirement without ambiguities, conflicts, misunderstanding and incompleteness. The specification further consists of following steps;

Step 1: User reviews about project management

The requirements of new projects have been gathered from different stakeholders using online web form, and interviews. In GSD, different stakeholders are participating from different locations and different organizational cultures in developing software. Therefore, for requirement elicitation different steps have been performed and documented for further analysis. These steps are described in Fig. 3.

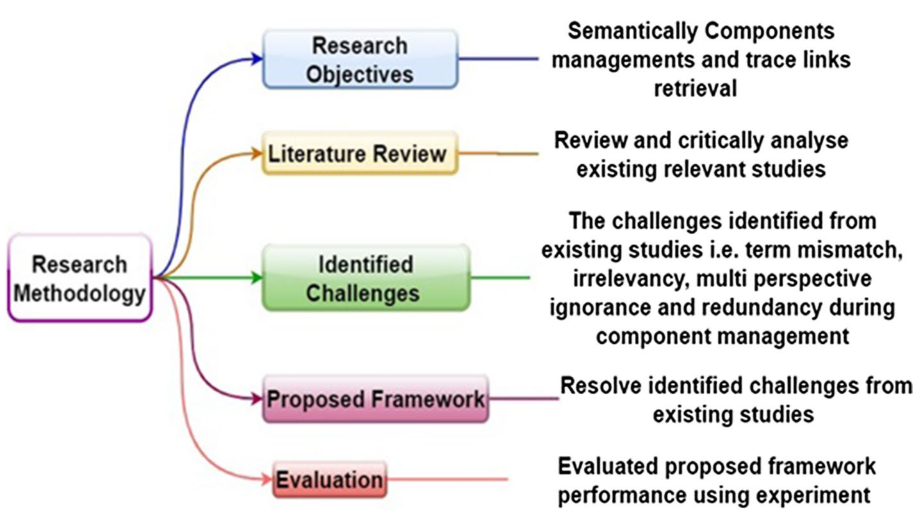

Fig. 1 Research methodology

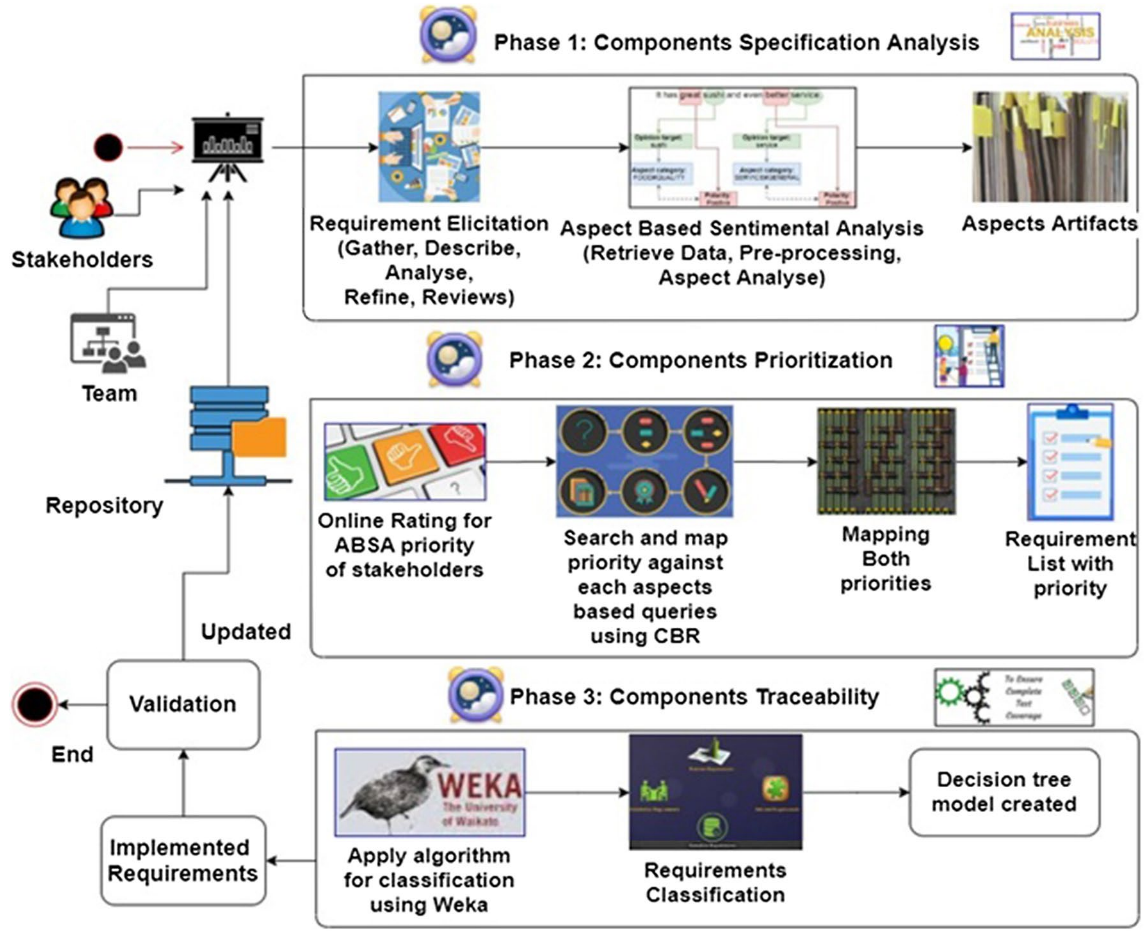

Fig. 2 CRMGDE framework overflow

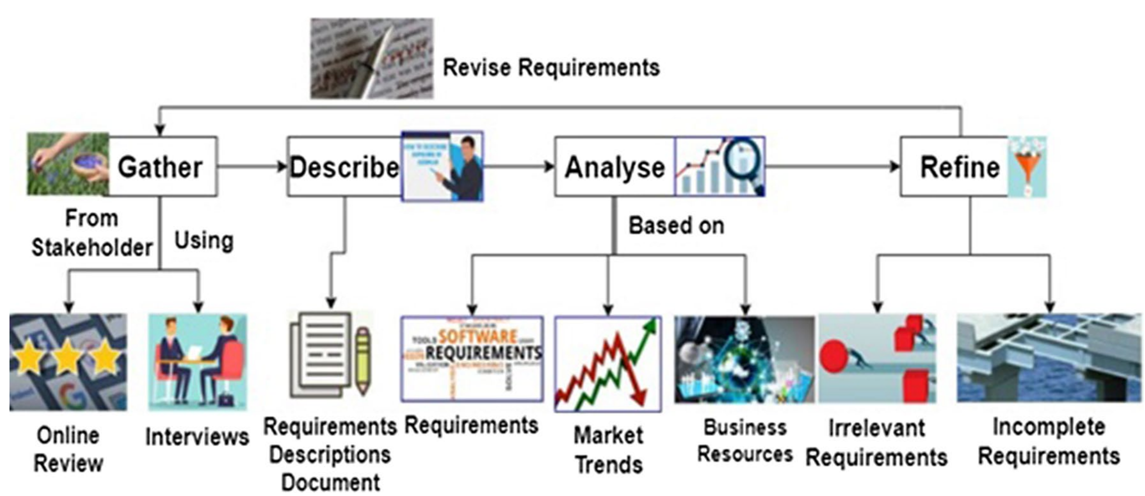

Fig. 3 Requirement elicitation process

The elicitation process depicted in Fig. 3 described collection of requirements for new or updating of software. Its start from gathering requirements from stakeholders of software using various techniques. Subsequently, ended refinement of requirement before specification, prioritization and other development processes. All the steps of elicitation process and their output monitored and updated in team foundation server (TFS) Repository to improve collaboration among development team and stakeholders. As, TFS has been used to resolve communication, coordination, and control problem in GSD environment to provide single platform for stakeholder, project management team and product development team for monitoring and managing continuous progress of product.

As most of products are based on various versions and family of products with combination of different components and modules. Therefore, all the evolution in software version and their components are core assets and variabilities to increase satisfaction level of their users according to user reviews, experiences and opinions. Thus, after requirements elicitation reviews of similar product extracted online forum linked with TFS for better product quality. For example;

> *If customer demand for evolution in Biometric device for university purposes. Then extracted requirements using elicitation method in the form of interviews and by filling online web forms. The reviews and experiences about previous versions and similar domain products extracted from repository using online forums. These comments may be positive, negative, neutral etc. and extracted from following types of information's (I);*

- I_1: The best device and great features without no errors in working.
- I_2: The device works flawless and its one year to use without any damage cost.
- I_3: The device works sometime required multiple try to match finger or thumb impression due to wetting'.

Step 2: Requirements Extraction using ABSA

After requirement specification extraction, different aspects-based sentiments have been analyzed as described in Fig. 4.

Aspect-based sentiment analysis extends terms extraction using sentiment analysis to further categories based on aspects of feature relevant to product. For example, a "car," feature has been categorized using its distinct aspects that is design and engine aspects.

Hence, the Apache OpenNLP sentence detector tool available for sentence separating. Subsequently, second step of ABSA tool to Tokenizing and parts of speech (POS) tagging (especially, in case of sentence start with adjective for accurate tags generation and occasionally tags generate as noun). Thirdly, all capital letter converted into small letters and then assign relevant aspects to these small letters. Fourthly, tool detect, and classify these aspects in accord to opinions and afterward, linked these aspects to their associated aspects.

Therefore, we categorized requirements with their frequency based on different stakeholders' viewpoints i.e. management, end user, financial, development etc. using ABSA. The example of ABSA is described in Table 2. Then all information's i.e. I_1, I_2, and I_3 is categorized into different sections for semantic analysis with sentiments using ABSA.

Table 2 ABSA working

Information	Aspects	Subcategory	Opinions
I_1	Product	Features	Positive
I_2	Cost	Maintenance cost	Positive
I_3	Services	Impression scanner	Negative

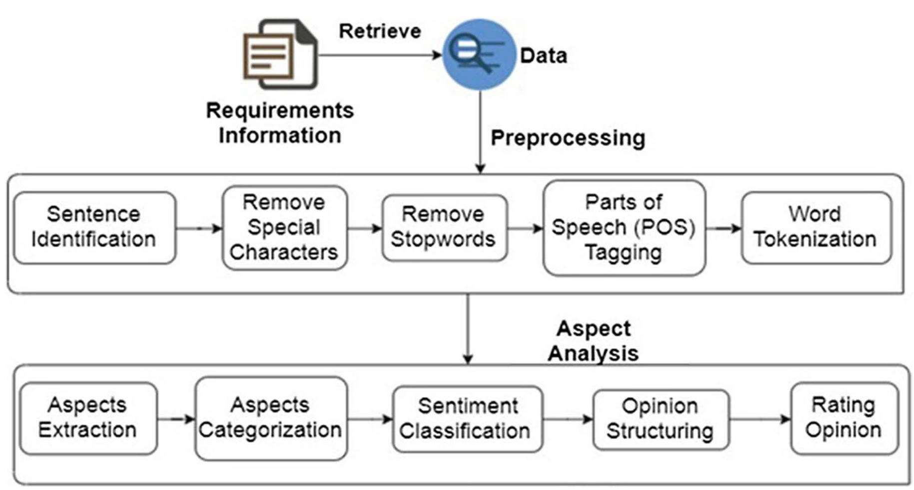

Fig. 4 Aspects analysis

3.1.2 Prioritization

After extraction of terms/features based on some aspects then getting rating of these features from stakeholders using TFS. These requirements are classified according to different aspects like Product, Costs, Services, Conditions etc. Due to this, requirement among stakeholders is divided according to aspects to reduce ambiguity and effort in rating requirements for implementation. For rating of requirements to implement most relevant and important features first to reduce redundancy and irrelevancy among requirements managements, following steps have been adopted.

For example, the aspect-based requirements are user Id with name and designation, finger and thumb scanner, sound system, face detector, check in, check out etc. These aspects are divided among relevant stakeholders to rank requirement from 1 to 5 scale according to importance. The relevant stakeholders are; Product Owner (PO), End User (EU), Administrative Staff (AS), and Development Team (DT) as described in Table 3.

As results described that some of the requirements have been missed due to diverse review or irrelevancy. The missing requirements may increase the chances of system failures. Therefore, to increase accuracy of priority, missing value have been identified from historical information using CBR technique.

Table 3 Online ranking

Req ID	Aspects	Sub aspects	Stakeholder	Ranking
R1	Features (F)	User Id (UI)	PO	2
			EU	2
			AS	4
			DT	3
R2	Features (F)	Finger scanner (FS)	PO	–
			EU	3
			AS	4
			DT	3
R3	Features (F)	Face detector (FD)	PO	4
			EU	–
			AS	4
			DT	3
R4	Services (S)	Check in (CI)	PO	3
			EU	2
			AS	–
			DT	2
R5	Services (S)	Check out (CO)	PO	2
			EU	3
			AS	–
			DT	3

Query based association using CBR for similar case priority

The CBR applied on the elicited requirements and retrieve from TFS based on stakeholders' queries for accurate priority extracted from previous similar requirements priority. The CBR procedure based on expert's knowledge and expertise to reduce time, cost and efforts intelligently. CBR reused priority of previously developed projects requirements and retrieved information from repository for new project similar requirements (as shown in Fig. 5).

In case if no exact similar case available then search for partial similar case otherwise involve stakeholders for missing requirements priority. It reduces complexity, inaccuracy, incompleteness, stakeholder's involvement and team efforts within limited available resources.

After the implementation of CBR technique, ranking has been identified based on previous similar cases ranking to identify missing requirements. The description of example results explains in Table 4.

Mapping for missing requirements priority

The matrix is created to find priority of missing requirements by mapping current and previous priorities. In matrix requirements are prioritized to identify accurate priority e.g. If login and password are two requirements of new system to develop which have been prioritized by stakeholders. The stakeholders prioritized only login requirement and consider password as part of login requirement but according to development team it is a separate requirement and need priority. Therefore, to reduce this ambiguity, team extract missing requirements priority which have been extracted in CBR process and in online rating according to stakeholder

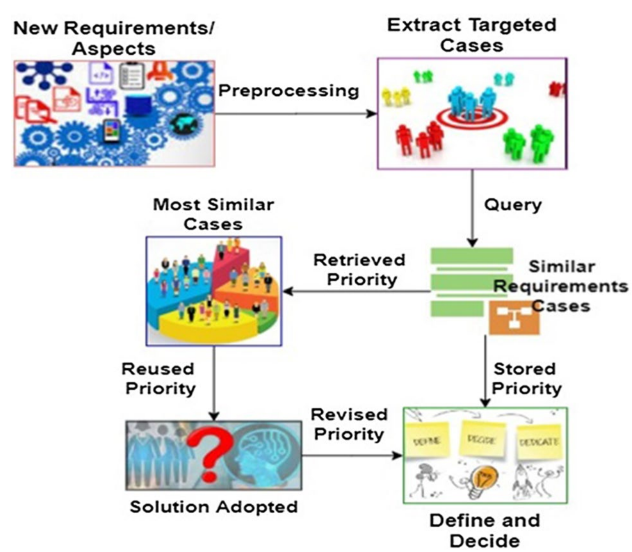

Fig. 5 Case-based reasoning process

Table 4 Ranking based on CBR

Req ID	Queries	Stakeholder	Product type	Ranking	Expert decision
R1	UI	PO	University	3	N
			Office	5	S
			Hospital	2	N
		EU	University	2	S
			Office	1	N
			Hospital	2	N
		AS	University	3	S
			Office	2	N
			News agency	2	N
		DT	University	3	S
			Office	3	N
			Hospital	3	N
			News agency	3	N
R2	FS	PO	University	2	N
			News agency	3	S
		EU	Office	2	N
			University	3	S
			News agency	2	N
		AS	University	3	N
			News agency	4	S
		DT	University	3	S
			News agency	3	N
			Hospital	3	N
R3	FD	PO	Government organization	3	S
			News agency	2	N
		EU	Media organization	4	S
			Office	3	N
		AS	Media organization	3	S
		DT	Government organization	2	N
			News agency	3	S
			Hospital	2	N

S Selected, *N* Not selected

Table 5 Mapped ranking

Req ID	Queries	Stakeholder	CBR Ranking	Current Ranking	Merge Ranking	New Ranking
R1	UI	PO	5	2	5	3.5
		EU	2	2	2	
		AS	3	4	4	
		DT	3	3	3	
R2	FS	PO	3	–	3	3
		EU	3	2	3	
		AS	3	1	3	
		DT	3	3	3	
R3	FD	PO	3	2	3	3.75
		EU	4	–	4	
		AS	3	5	5	
		DT	3	3	3	

perspective. Table 5 described the new priority after merging CBR based priority and online priority of software.

3.2 Traceability

To monitor and validate requirements trace links which were generated from requirements to requirements throughout the development process using machine learning technique, thus it identifies and classify requirements according to their dependency among other requirements to check functionality responsibilities or coupled with each other for different perspective. It helps to implement changes in requirements of product. The requirement traceability matrix (RTM) maps requirements to requirements relationship which describe their dependency among each other;s. To verify and validate change analysis, reuse requirements and regression using RTM which allows prediction of changes and their relevant impact on whole system.

Thus, for classification of requirements classification, algorithm of Weka tool has been used after prioritization process.

The Decision tree classifier based on J48 algorithm in Weka tool has been adopted for classification of these requirements. It has been used to create tree of requirements start from root nodes and divided into different leaves and sub leave until all requirement relation or dependency is identified. It makes learning by applying machine learning algorithm which is efficient, error free, and simple. Results have been represented graphically with statistical results which are able to identify error, incompleteness and ambiguities in data. Thus, it is useful for verifying and validating accuracy of aspects and their priority identified in previous phases. Then further implementation with less chances of failure and able to easily apply changes without error in system.

4 Results and discussions

The CRMGDE framework implemented for performance evaluation in real-world context using different datasets have been conducted through an experimental approach. In experiment two types of participants have been included; one type of participants has (TP1) adopted RMDGE framework and second type of participants has (TP2) adopted other technique without ABSA and CBR.

For performance evaluation of CRMDGE three projects of technology development organization have been taken which are working in distributed environment and have their stakeholder globally. The organization not allowed to disclosed their complete information due to privacy issue and selected two projects datasets i.e. Dataset 1 (*D*1): LMS system; and Dataset 2 (*D*2): Card swipe machine.

The data information through elicitation and previous reviews about datasets have been collected and have been done through all steps of CRMDGE framework and without CRMDGE framework as well. The CRMDGE framework proposed to increase user/stakeholder's satisfaction, accurate priority and correct relationship of requirements to reduce failures rate within limited resources. To extract satisfaction, questionnaire based data collection process has been adopted based on some parameters extracted from existing literature that are; easiest to adopt (EA), Improve Management Process (IMP), Improve Distributed Issues (IDI), Sentimental Analysis useful (SAU), Human Effort Reduce (HER), Reduce Irrelevancy and redundancy (RIR), Completeness of requirement Increases (CRI), Improve Knowledge Management (IKM), Reduce requirement inaccuracy (RRI), Reduce stakeholder participation (RSP), Accurate Extraction Aspects (AEA), and Increases Reusability Requirement (IRR). The satisfaction level has been identified from questionnaire analysis. The analysis based on three scales point i.e. positively (*P*), negatively (*N*) and affectless (*A*).

The CRMGDE framework results compared with without CRMGDE methods i.e. previous selection frequency and expert based. In previous selection frequency method participants select and prioritize components based on last version components priority. While in expert-based method components management process depends on the decision of experts and experts selected based on knowledge and experience.

The Fig. 6 describes the satisfaction level of *P* scale point among both types of participants. The *x*-axis describes the satisfaction of participants in terms of percentage and *y*-axis describes the factors used in questionnaire construction. The results of *P* depicted that more than fifty percent participants positively reply to questionnaire and improvement in software quality.

The results of *N* depicted in Fig. 7 that less than fifty percent participants negatively replied to questionnaire and improvement in software quality. The participants

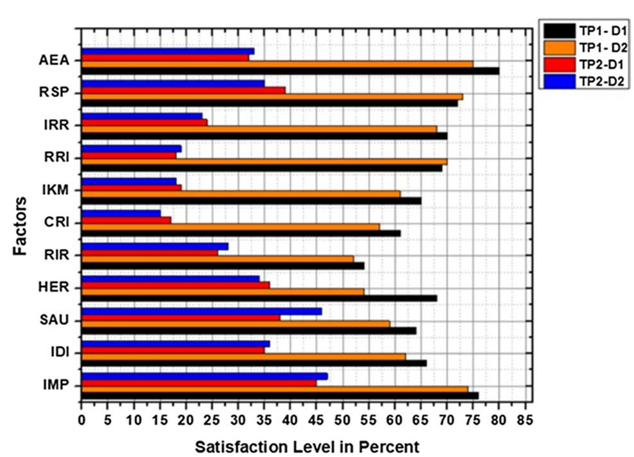

Fig. 6 Review analysis of *P*

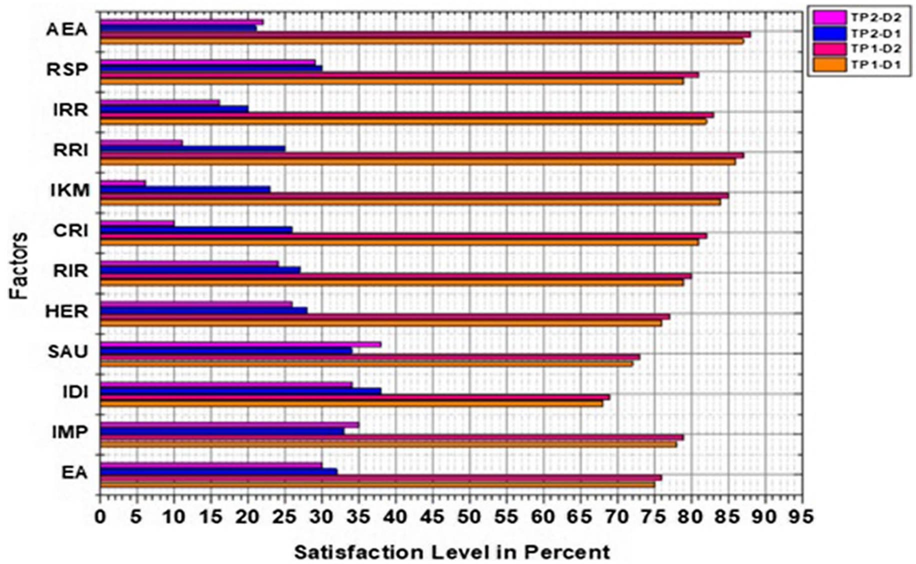

Fig. 7 Review analysis of *N*

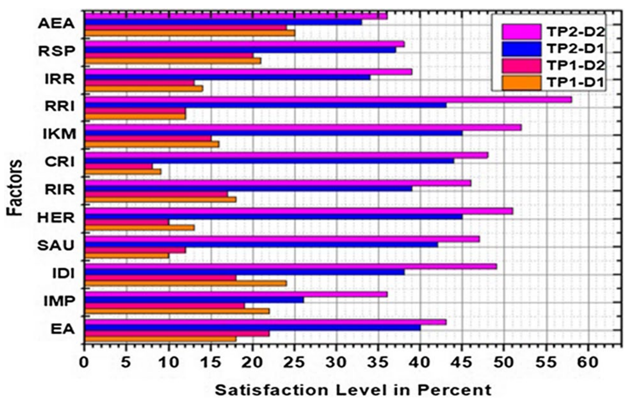

Fig. 8 Review analysis of *A*

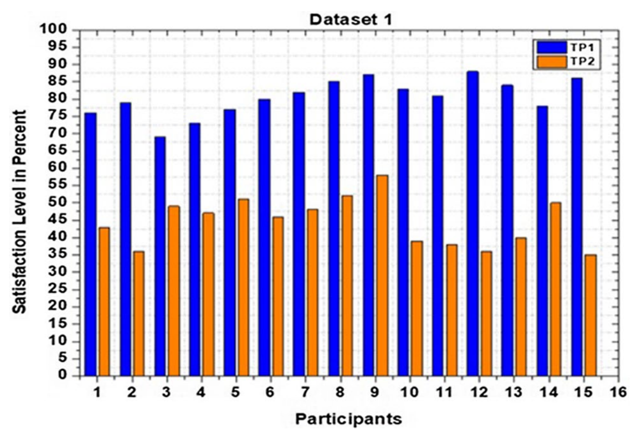

Fig. 9 Comparative review of dataset 1

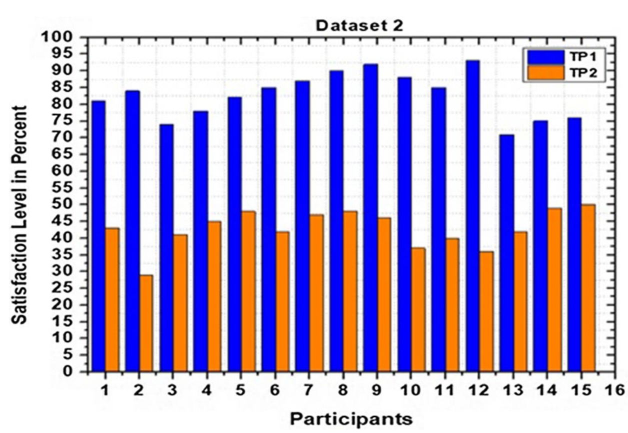

Fig. 10 Comparative review of dataset 2

of CRMGDE in all datasets have approved that most of the participants have higher satisfaction level then without CRMGDE (WO-CRMGDE) method. Thus, ABSA of requirements with CBR technique and requirements trace links using decision tree classification of requirements improved the quality and reduce ambiguities throughout of the development process.

Whereas, Fig. 8 answers the questions based on *A* and describes that all parameters improve the performance of CRMGDE using two different datasets. Few of the participants have no impact or neutral point of view about the performance of CRMGDE as compare to WO-CRMGSE. The view point may be differed due to expertise and prior knowledge about the approaches, but overall review analysis described that CRMGDE efficiently removes existing studies limitations for higher quality.

The overall comparative analysis of approaches i.e. CRMGDE and WO-CRMGDE is shown in Fig. 9 for *D1* and Fig. 10 for *D2* to analyze the satisfaction level participants wise. The *x*-axis describes the TP1 and TP2 satisfaction level in percent and *y*-axis describes numbers of participants both in TP1 TP2.

The comparative review analysis of both datasets explains that the satisfaction level of *TP1* is more than 50 percent using CRMGDE as compare to WO-CRMGDE which have less than 50 percent *TP2* satisfaction level. The experimental comparative analysis proves that the selected factors are able to improve development process and product quality.

To measure the performance and accuracy of requirements classification using CRMGDE, different measures have been performed and compared with WO-CRMGDE method. These measures are Precision which shows the ratio between correctly linked requirement to total number of requirements linked after ABSA and prioritization, Recall that describes the ratio between correctly linked and total number of requirements after ABSA and prioritization and finally *F*-Measure which uses combination of both recall and precision using equation (1) [28, 34, 41].

$$F_{\text{Measure}} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (1)$$

The values are measured in percentage and from 0 to 100 percent, more near to 100 percent shows more accurate CRMGDE performed and vice versa. The detail classification analysis of performance measures is described in Table 6.

The graphical analysis of performance measures is described in Fig. 11. The x-axis and y-axis describe the value in percentage and approaches with datasets details respectively. All measure values near to 100 percent in case of CRMGDE in both datasets, depicts that CRMGDE significantly improved the requirement management process. While in WO-CRMGDE case all measure values near to zero percent in both datasets depicts that ABSA, CBR and classification are important for requirement management process.

The experimental results proved that the CRMGDE framework is beneficial for requirement management process and for higher software quality using ABSA in globally distributed environment. During evaluation we found that requirement change management process has been improved with accurate classification of requirements based on semantics.

4.1 Threats to validity (TV)

For empirical evaluation, some threats occur that argument hypothetical rationality of the outcomes. This requires duplication of investigation to accept/disprove verdicts. The essential threats are; internal TV (ITV), external TV (ETV), construct TV (CTV), and reliability TV (RTV).

ITV associated to aspects concerning procedure of component management process. To report this threat, we extenuation steps implemented to avoid using diverse measures for CBS managing activities. And results proved that CRMGDE improves prioritization and traceability process. ETV narrates to generalization of findings in four real projects in contrast to used example for assessment. This enhances the validity of results by replicating the *PF* steps in diverse situations.

CTV reflects the association among numerous perceptions and reflections. This initiated use of different measures to evaluate CTV of various procedures i.e. aspects analysis, prioritization and traceability in CRMGDE and to estimate performance as linked to other technique. RTV narrates relations between action and consequence. This can be mitigated by using rigorous real-world calculation in CRMGDE for verification with all authors involvements during data collection and analysis. Therefore, to avoid RTV we used to experiment for evaluating the learning effect may all have influenced the results to reduce biasness.

5 Conclusion and future works

The semantic, similar case analysis and requirements classification helps in increasing stakeholder satisfaction level and quality of product with reduction in irrelevancy, redundancy and stakeholder's involvement. For semantic analysis, to analyze stakeholder diverse perspective with their less involvement using aspect based sentimental analysis, helps in the improvement of components specification process of requirements. To improve components prioritization process, similar cases-based technique has been utilized to identify accurate priority without any ambiguities. For the similar case priority extraction, case-based reasoning

Table 6 Comparisons of classification measures

Measures	D1		D2		D3	
	CRMGDE	WO-CRMGDE	CRMGDE	WO-CRMGDE	CRMGDE	WO-CRMGDE
Precision	86.84	81.25	86.55	80.85	84.74	81
Recall	82.9	78.53	82.69	73.69	83.89	76.65
F-measure	84.83	78.8	85.1	77.12	84.31	78.76

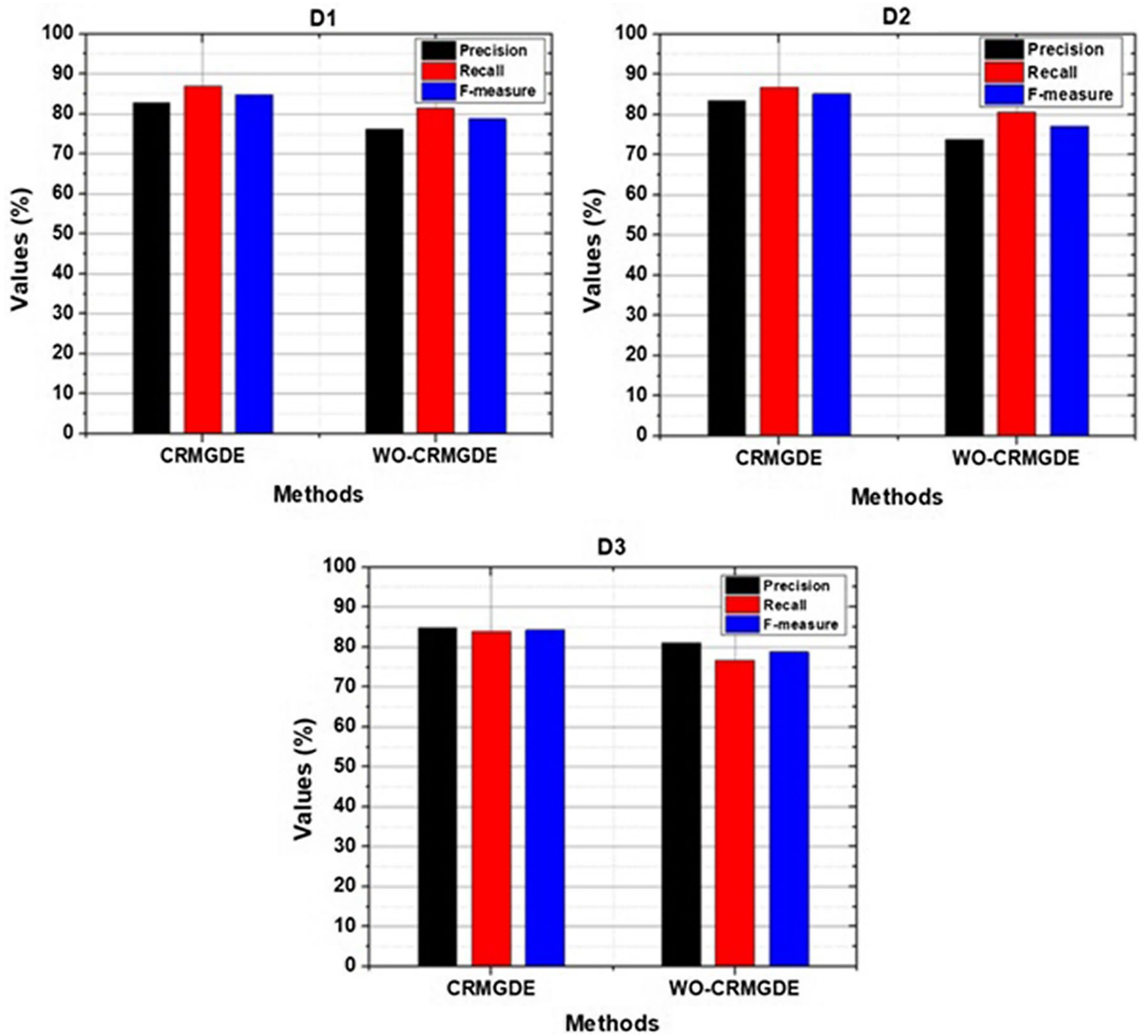

Fig. 11 Measures comparison

method has been applied to use historical information for identifying priority of missing requirements which also help to precise the priority of tied requirements which have same current priority. For traceability requirements, classification based on decision tree has been used to verify and validate requirements as specification, prioritization and traceability are the important phases of requirements management process during product development cycle. It has been identified from literature that management process become more complicated due to large number of stakeholders and distributed location. Thus, based on these concepts' management of components requirement framework for globally distributed environment have been proposed to improve the quality of product. To evaluate the performance of proposed framework, experimental approach has been adopted on two different datasets. Different measures have been applied to evaluate the accuracy of aspects, priority and trace links and results represent noteworthy improvement. The proposed framework outperformed when compared with other existing approaches. The results motivate us to continue the research with the context to manage component change in requirements and validating requirements by enhancing the proposed framework. The framework could be used to predict error prone requirements based on historical information and sentiment analysis.

In future , we are planning to enhance proposed framework using aspect based sentimental analysis for the selection of components, prioritization of configurable systems and verification of configuration after changes in components..

Acknowledgements This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under Grant Number 102.03-2019.10.

References

1. Al-Hajjaji M, Lity S, Lachmann R et al (2017) Delta-oriented product prioritization for similarity-based product-line testing. In: Proceedings of the 2017 IEEE/ACM 2nd international workshop on variability and complexity in software design (VACE). IEEE, Buenos Aires, Argentina, pp 34–40
2. Ali N, Cai H, Hamou-Lhadj A, Hassine J (2019) Exploiting parts-of-speech for effective automated requirements traceability. Inf Softw Technol 106:126–141. https://doi.org/10.1016/j.infsof.2018.09.009
3. Ali S, Hafeez Y, Jhanjhi NZ et al (2020) Towards pattern-based change verification framework for cloud-enabled healthcare component-based. IEEE Access 8:148007–148020. https://doi.org/10.1109/ACCESS.2020.3014671
4. Alsanad AA, Chikh A, Mirza A (2019) A domain ontology for software requirements change management in global software development environment. IEEE Access 7:49352–49361. https://doi.org/10.1109/ACCESS.2019.2909839
5. Alzyoudi R, Almakadmeh K, Natoureah H (2015) A probability algorithm for requirement selection in component-based software development. Acm Ipac 15:1–6. https://doi.org/10.1145/2816839.2816871
6. Arias M, Buccella A, Cechich A (2018) A framework for managing requirements of software product lines. Electron Notes Theor Comput Sci 339:5–20. https://doi.org/10.1016/j.entcs.2018.06.002
7. Asif SA, Masud Z, Easmin R, Gias AU (2017) SAFFRON: a semi-automated framework for software requirements prioritization. Int J Adv Comput Sci Appl 8:10.14569/IJACSA.2017.081265
8. Ayala C, A methodology to rank the design A, Franch X, et al (2018) System requirements-OSS components: matching and mismatch resolution practices: an empirical study. Empir Softw Eng 23:3073–3128
9. Babar MI, Ghazali M, Jawawi DNA et al (2015) PHandler: an expert system for a scalable software requirements prioritization process. Knowl Based Syst 84:179–202. https://doi.org/10.1016/j.knosys.2015.04.010
10. Barbosa PAM, Pinheiro PR, Silveira FRV, Filho MS (2019) Selection and prioritization of software requirements applying verbal decision analysis. Complexity 2019:1–20. https://doi.org/10.1155/2019/2306213
11. Bashir I, Hamid B, Jhanjhi NZ, Humayun M (2020) Systematic literature review and empirical study for success factors: client and vendor perspective. J Eng Sci Technol 15(4):2781–2808
12. Bashir U, Chachoo M (2017) Performance evaluation of j48 and Bayes algorithms for intrusion detection system. Int J Netw Secur Appl IJNSA 9
13. Bjarnason E (2013) Distances between requirements engineering and later software development activities: a systematic map. In: Doerr J, Opdahl AL (eds) Requirements engineering: foundation for software quality. Springer, Berlin, pp 292–307
14. Borg M, Chatzipetrou P, Wnuk K et al (2019) Selecting component sourcing options: a survey of software engineerings broader make-or-buy decisions. Inf Softw Technol 112:18–34. https://doi.org/10.1016/j.infsof.2019.03.015
15. Borrego G, Morán AL, Palacio RR et al (2019) Towards a reduction in architectural knowledge vaporization during agile global software development. Inf Softw Technol 112:68–82. https://doi.org/10.1016/j.infsof.2019.04.008
16. Carrillo de Gea JM, Nicolás J, Fernández Alemán JL et al (2016) Co-located and distributed natural-language requirements specification: traditional versus reuse-based techniques: co-located and distributed natural-language requirements specification. J Softw Evol Process 28:205–227. https://doi.org/10.1002/smr.1772
17. Chatzipetrou P, Papatheocharous E, Wnuk K et al (2019) Component attributes and their importance in decisions and component selection. Softw Qual J 28:1–27
18. Cleland-Huang J, Gotel OCZ, Huffman Hayes J, et al (2014) Software traceability: trends and future directions. In: Proceedings of the on Future of Software Engineering—FOSE 2014. ACM Press, Hyderabad, pp 55-69
19. Costa MCDC, Lemos GS, Beck F (2018) Software engineering tools environment for outsourcing teams collaboration. IEEE/ACM 13th international conference on global software engineering (ICGSE)
20. Cuzzocrea A, Pilato G (2019) An innovative user-attentive framework for supporting real-time detection and mining of streaming microblog posts. Soft Comput. https://doi.org/10.1007/s00500-019-04478-2
21. Dekhtyar A, Hayes JH (2018) Automating requirements traceability: two decades of learning from KDD. ArXiv180711454 Cs
22. Dvořák O, Pergl R, Kroha P (2017) Tackling the flexibility-usability trade-off in component-based software development. In: Rocha Á, Correia AM, Adeli H et al (eds) Recent advances in information systems and technologies. Springer, Cham, pp 861–871
23. Fernández-García AJ, Iribarne L, Corral A et al (2019) A recommender system for component-based applications using machine learning techniques. Knowl Based Syst 164:68–84. https://doi.org/10.1016/j.knosys.2018.10.019
24. Gambo I, Ikono R, Achimugu P, Soriyan A (2018) An integrated framework for prioritizing software specifications in requirements engineering. Int J Softw Eng Its Appl 12(33–46):10.14257/ijseia.2018.12.1.03
25. Guo J, Gibiec M, Cleland-Huang J (2017) Tackling the term-mismatch problem in automated trace retrieval. Empir Softw Eng 22:1103–1142. https://doi.org/10.1007/s10664-016-9479-8
26. Gupta A, Gupta C (2019) A novel collaborative requirement prioritization approach to handle priority vagueness and inter-relationships. J King Saud Univ Comput Inf Sci. https://doi.org/10.1016/j.jksuci.2019.12.002
27. Hafeez Y, Asghar S, Ahmad A, Hussain A (2016) A taxonomy to support team coordination in global software engineering environment. J Comput Theor Nanosci 13:3238–3247. https://doi.org/10.1166/jctn.2016.4981
28. Hatipoglu PU, Atvar A, Artan YO et al (2017) Software requirement traceability analysis using text mining methods. In: Proceedings of the 2017 25th signal processing and communications applications conference (SIU). IEEE, Antalya, Turkey, pp 1–4
29. Hong Y, Kim M, Lee S-W (2010) Requirements management tool with evolving traceability for heterogeneous artifacts in the entire life cycle. In: Proceedings of the 2010 eighth ACIS international conference on software engineering research, management and applications. IEEE, Montreal, QC, Canada, pp 248–255
30. Humayun M, Jhanjhi NZ (2019) Exploring the relationship between GSD, knowledge management, trust and collaboration. J Eng Sci Technol 14(2):820–843
31. Kamal T, Zhang Q, Akbar MA (2020) Toward successful agile requirements change management process in global software development: a client-vendor analysis. IET Softw 14:265–274. https://doi.org/10.1049/iet-sen.2019.0128
32. Lahon, M, Uzzal S, International Conference on Computing for Sustainable Global Development (2016) Risk assessment and mitigation approach for architecture evaluation in component based software development. In: Proceedings of the 10th INDIACom;

2016 3rd international conference on computing for sustainable global development: (16th–18th March, 2016) INDIA Com-2016
33. Lai R, Ali N (2013) A requirements management method for global software development. AIS Adv Inf Sci 1:38–58
34. Li Y, Yue T, Ali S, Zhang L (2019) Enabling automated requirements reuse and configuration. Softw Syst Model 18:2177–2211. https://doi.org/10.1007/s10270-017-0641-6
35. Liu G, Huang X, Liu X, Yang A (2020) A novel aspect-based sentiment analysis network model based on multilingual hierarchy in online social network. Comput J 63:410–424. https://doi.org/10.1093/comjnl/bxz031
36. Lu T, Liu C, Duan H, Zeng Q (2020) Mining component-based software behavioral models using dynamic analysis. IEEE Access 8:68883–68894. https://doi.org/10.1109/ACCESS.2020.2987108
37. Mohan A, Jha SK (2019) Predicting and accessing reliability of components in component based software development. In: Proceedings of the 2019 international conference on intelligent computing and control systems (ICCS). IEEE, Madurai, India, pp 1110–1114
38. Musto C, de Gemmis M, Semeraro G, Lops P (2017) A multi-criteria recommender system exploiting aspect-based sentiment analysis of users reviews. In: Proceedings of the eleventh ACM conference on recommender systems. ACM, Como Italy, pp 321–325
39. Nicolas J, Carrillo De Gea JM, Nicolas B et al (2018) On the risks and safeguards for requirements engineering in global software development: systematic literature review and quantitative assessment. IEEE Access 6:59628–59656. https://doi.org/10.1109/ACCESS.2018.2874096
40. Olaronke I, Rhoda I, Ishaya G (2018) An appraisal of software requirement prioritization techniques. Asian J Res Comput Sci. https://doi.org/10.9734/ajrcos/2018/v1i124717
41. Pereira JA, Matuszyk P, Krieter S, et al (2016) A feature-based personalized recommender system for product-line configuration. In: Proceedings of the 2016 ACM SIGPLAN international conference on generative programming: concepts and experiences—GPCE 2016. ACM Press, Amsterdam, Netherlands, pp 120–131
42. Perini A, Susi A, Avesani P (2013) A machine learning approach to software requirements prioritization. IEEE Trans Softw Eng 39:445–461. https://doi.org/10.1109/TSE.2012.52
43. Pitangueira AM, Maciel RSP, Barros M (2015) Software requirements selection and prioritization using SBSE approaches: a systematic review and mapping of the literature. J Syst Softw 103:267–280. https://doi.org/10.1016/j.jss.2014.09.038
44. Port D, Nikora A, Hayes JH, Huang LG (2011) Text mining support for software requirements: traceability assurance. In: Proceedings of the 2011 44th Hawaii international conference on system sciences. IEEE, Kauai, HI, pp 1–11
45. Riaz MQ, Fateh-ur-Rehman MB, Butt WH (2018) Customization of requirement engineering best practices for Pakistan software industry. In: Proceedings of the 2018 International conference on computing, mathematics and engineering technologies (iCoMET). IEEE, Sukkur, pp 1–6
46. Santos R, Albuquerque A, Pinheiro PR (2016) Towards the applied hybrid model in requirements prioritization. Proc Comput Sci 91:909–918. https://doi.org/10.1016/j.procs.2016.07.109
47. Shameem M, Khan AA, Hasan MG, Akbar MA (2020a) Analytic hierarchy process based prioritisation and taxonomy of success factors for scaling agile methods in global software development. IET Softw 14:389–401. https://doi.org/10.1049/iet-sen.2019.0196
48. Shameem M, Kumar RR, Nadeem M, Khan AA (2020b) Taxonomical classification of barriers for scaling agile methods in global software development environment using fuzzy analytic hierarchy process. Appl Soft Comput 90:106122. https://doi.org/10.1016/j.asoc.2020.106122
49. Sharma S, Pandey SK Integrating AI techniques in requirements phase: A. Int J Comput Appl 975
50. Smiti A, Elouedi Z (2019) Dynamic maintenance case base using knowledge discovery techniques for case based reasoning systems. Theor Comput Sci. https://doi.org/10.1016/j.tcs.2019.06.026
51. Trainer EH, Redmiles DF (2018) Bridging the gap between awareness and trust in globally distributed software teams. J Syst Softw 144:328–341. https://doi.org/10.1016/j.jss.2018.06.028
52. Veer Singh Y, Kumar B, Chand S (2019) A hybrid approach for requirements prioritization using LFPP and ANN. Int J Intell Syst Appl 11:13–23. https://doi.org/10.5815/ijisa.2019.01.02
53. Wieringa R (2017) Requirements engineering since the year one thousand. In: Proceedings of the 2017 IEEE 25th international requirements engineering conference (RE). IEEE, Lisbon, Portugal, pp 480–481
54. Winkler S, von Pilgrim J (2010) A survey of traceability in requirements engineering and model-driven development. Softw Syst Model 9:529–565. https://doi.org/10.1007/s10270-009-0145-0
55. Wu D, Li J, Bao C (2018) Case-based reasoning with optimized weight derived by particle swarm optimization for software effort estimation. Soft Comput 22:5299–5310. https://doi.org/10.1007/s00500-017-2985-9
56. Zainuddin N, Selamat A, Ibrahim R (2016) Improving twitter aspect-based sentiment analysis using hybrid approach. In: Nguyen NT, Trawiński B, Fujita H, Hong T-P (eds) Intelligent information and database systems. Springer, Berlin, pp 151–160
57. Zawawi N, Hamdy M, Ghary R, Tolba MF (2016) Realization of a data traceability and recovery service for a trusted authority service co-ordination within a Cloud environment. Soft Comput 20:5039–5050. https://doi.org/10.1007/s00500-015-1792-4
58. Zhao L, Zhao A (2019) Sentiment analysis based requirement evolution prediction. Fut Internet 11(2):52. https://doi.org/10.3390/fi11020052