Research Article

Rao Nadeem, Rana M. Amir Latif, Khalid Hussain, N. Z. Jhanjhi*, and Mamoona Humayun

A flexible framework for requirement management (FFRM) from software architecture toward distributed agile framework

https://doi.org/10.1515/comp-2022-0239
received March 9, 2020; accepted April 4, 2022

Abstract: GSD is a Global software development environment where data are distributed to more than two sites. These sites may be located on national or continental borders. Every year it expands its development volume, which has become a trend for the software development business. Because of increasing demand, it has become a common type of business. The value of offshore software development has increased immensely. According to the recent predictions, one-quarter of US software-related business shifts offshore, including integration and management-related services. GSD helps explore resources from other countries, increasing knowledge and enhancing operational efficiency. Therefore, globalization has changed the development nature of software. Different organizations face several challenges, such as coordination and requirement ambiguity, during the change management process in GSD. Traceability accommodates these changes in forward and backward direction. However, it gives rise to several challenges like less client involvement because of its distributed nature and challenging to manage the requirement ambiguity due to increased cost. Therefore, the Flexible Framework for Requirement Management (FFRM) must handle the abovementioned issues.

Keywords: global software development, flexible framework for requirement management, agile framework, software engineering, traceability

1 Introduction

The co-located projects require working in an organized form and sharing different points of view related to the work. Through constant cooperation, team members have a clear idea of the project's nature, know the type of expertise, and allocate responsibilities according to their expertise [1]. All the team members know the progress of work and know how their work affects other people's tasks. The situational method is used in such an environment where requirements are changing. According to the scenario, a situational method is used as a base model and more functionality [2]. Many possible advantages can be achieved from global software development (GSD). One of the most significant advantages is reduced software development time because the work is divided between different locations according to expertise. Software development faces several challenges in a distributed environment, including time, culture, space, and communication gaps [3]. Scrum is commonly applied in GSD to understand customers and developers better. Scrum offers different characteristics to remove different aspects of culture and language-related problems [4]. Here we explain the GSD issues and one of the possible solutions by using agile practices in GSD. When software system complexity and sophistication increase, requirement analysis and management become a significant issue in a distributed environment [5]. These issues are also the same for multisite software projects, as shown in Figure 1.

*** Corresponding author: N. Z. Jhanjhi,** Department of Computer Science, Taylor's University, Lakeside Campus No. 1 Jalan Taylors, Subang Jaya, Selangor, Malaysia,
e-mail: noorzaman.jhanjhi@taylors.edu.my
Rao Nadeem: Department of Computer Science PMAS-Arid Agriculture University Rawalpindi, Rawalpindi, Pakistan,
e-mail: rao.nadeem96@gmail.com
Rana M. Amir Latif: Department of Computer Science, COMSATS University Islamabad, Sahiwal Campus, Rawalpindi, Pakistan,
e-mail: ranaamir10611@gmail.com
Khalid Hussain: Department of Cyber Security, National Institute of Science & Technology, Islamabad, Pakistan,
e-mail: kusmani.utm@gmail.com
Mamoona Humayun: Department of Computer and Information Sciences, Jouf University, Al-Jouf, KSA,
e-mail: mahumayun@ju.edu.sa

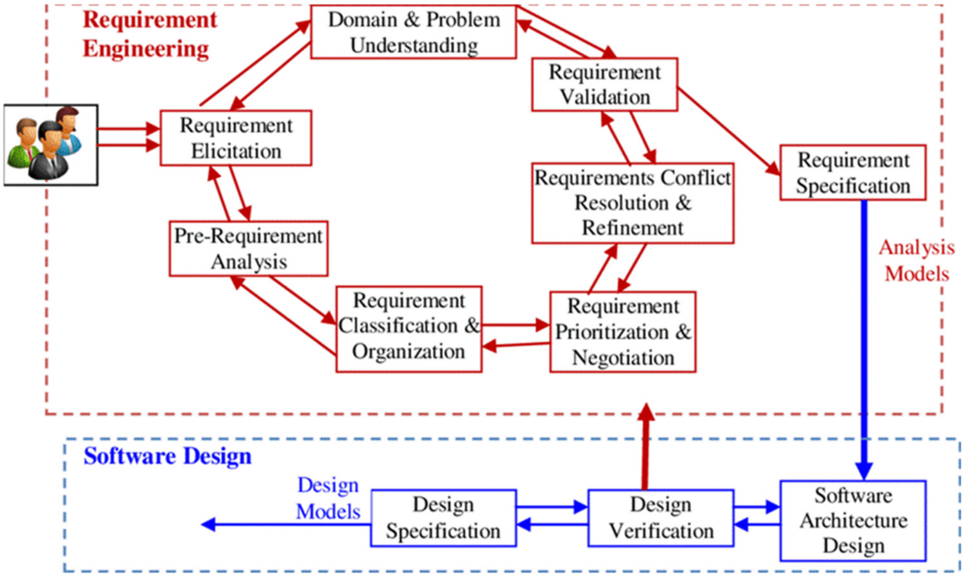

Requirement Engineering
Domain & Problem Understanding
Requirement Validation
Requirement Elicitation
Requirement Specification
Requirements Conflict Resolution & Refinement
Pre-Requirement Analysis
Analysis Models
Requirement Classification & Organization
Requirement Prioritization & Negotiation
Software Design
Design Models
Design Specification
Design Verification
Software Architecture Design

Figure 1: Requirement engineering phase.

It has elucidated the relevance of Requirements Management by introducing new approaches for dealing with global hazards in multi-site projects. This study's principal purpose is to provide a framework that provides the best practices for the enhancement of global initiatives. The primary objective of this research is to build a framework that would provide the best practices for the improvement of global projects [6]. This research is based on field study and has an industrial background. Data were gathered through surveys and interviews from experts and experienced project managers in a distributed environment. The data analysis is done according to principles prescribed in the research methodology [7]. Various types of artefacts, like requirements specification documents, model descriptions, source code, and test cases, are produced in Software development. These artefacts provide different views on the system at different points in time [8].

Traceability accommodates the change in both forward and backward directions. When change is implemented on distributed sites, it must be synchronized and well managed. Disconnected sites have high chances of destabilizing the project schedule and increasing costs [9]. Requirements tracing has essential benefits in the software development context, e.g., capturing traces in weekly intervals to acknowledge the development team whose requirements are implemented improves coordination [10]. The project manager keeps an eye on the interdependencies between several artefacts, requirement conflicts, understanding of change requests, and their impact on the quality of software that is essential for customer satisfaction [11].

Identification of requirement conflicts management, impact analysis, program comprehension, model consistency checking, release planning, verification, and validation (testing) are all approaches discussed in the literature. Although, when change occurs, increasing requirements require tracing precision, leading to more effort, making it expensive to identify and maintain trace dependencies [11]. Several tools are used for requirement traceability that creates the trace links and checks the requirements dependencies with high-quality results at minimum cost. The purpose of most tools is to automate requirements tracing, but automating requirement tracing is still complicated and an error-prone task. Moreover, requirement tracing requires considerable work and focus; only automation eliminates this need [12].

Changing requirements is not a significant issue, but the issue is how to handle this change on distributed sites where all the members are dispersed. The proposed model is evaluated using a survey based on experts with previous research. The primary purpose of requirement management (RM) is to validate the requirement statements and manage the documents so that the product delivery is correct and satisfies the user. Change implementation on different sites creates communication problems and increases the cost. Notification of the change implemented on one site is needed to be updated in all other sites. Several research papers explain the significance of RM and new practices to minimize requirement change management (RCM) issues. The system is not successful when it is complex. The guiding principle in the required category defines the traceability and consistency between the requirements and the unified modeling language (UML) model.

This research aims to improve communication and cost issues. We want to propose an interface that integrates all the sites and notifies the sites if any change occurs in any site. In this proposed model, we cover the communication and cost problems.

1.1 Literature review

An agile practitioner's viewpoint on trust is examined in this research via the prism of a multidimensional perspective. Organizations and software practitioners may use the findings to foster trust while implementing agile in a dispersed context [13].

Large-scale projects have used agile approaches to improve software delivery by teams. This article presents a design framework based on logical architectural models to distribute work items among distant agile teams. A study project shows how it provides a systematized strategy for decision assistance for agile teams based on requirement models and shared knowledge of the problem [14].

The high failure rate of distributed agile development can be reduced by deploying an agile enterprise architecture; however, the effectiveness of this approach is yet to be shown. After interviewing and attending 4 team sessions over 2 months, the data were analyzed using thematic analysis. However, many questions still need to be answered [15].

Large software businesses have difficulty allocating and managing work in a dispersed agile software development environment. When assigning work to a person or a team, many things must be considered. The most relevant papers from 2010 to 2019 were identified by a comprehensive examination of the scientific literature using Google Scholar. The article summarizes and discusses the review's findings [16].

In light of the rising prevalence of machine learning (ML) and the resulting complexity of today's software architectures, two new fields of study have emerged: software architecture for ML-based systems and ML for software architecture. To better define a standard set of practices for ML and software practitioners, the author identifies four critical areas of software architecture [17].

Moodle is a developing platform for Virtual Learning Management Systems that became a standard de facto for most educational institutions throughout the globe. A plug-in in Moodle may infer each student's learning style attending a course using an enhanced version of a Bayesian network model the author previously tested. Moodle aims to assist individualized instruction based on the Felder-ILS Silverman's model and tested via controlled trials and pilot tests in high schools and university courses [18].

The author explains the difference between situational method engineering (SME) projects and context type methods. In previous method engineering approaches, the context type method is ignored. To mitigate this issue, they identified the project type and context type factors to enhance some existing method engineering approaches by adding some steps [19]. They also proposed the Meta model that enables the method engineer to differentiate using situational methods. The method is a sequence to perform a system that gives the directions, rules, and structure to be followed in the development procedure. The primary purpose of this article is to present the state of art in a specific subset of design research called method engineering. The situational method extracts the base method regarding the project's environment and accommodates it according to the project situation [20].

Composite of the situational method (SM) is based on three phases (situational characteristics identifying, dividing generic artefacts into fragment artefacts, and artefact composing in SM). According to the previous research method of construction and implementation, there are five elements: design activities, specifying documents for design results, roles, method and techniques, and method information in the form of the model [21]. These five elements are added in metamodel [22].

In this article, the author explains the procedure to model with any RM methodology having four steps (planning and evaluation of method, project and context factors identification, project and context type analysis, and validation of SM) [23].

The author proposed an improved framework of RCM in GSD. The proposed solutions follow RCM processes and mitigate GSD issues. These phases are the initiation of change, evaluation of the change, voting process, and change implementation [24]. The change in the distributed environment becomes complicated because of the lack of communication and cooperation with globally dispersed stakeholders. Previous research work is compared with the proposed framework, and an analysis report is generated based on domain expert feedback. Evaluation of the framework shows that it fulfils the user requirements of RCM. The proposed model provides efficient and reliable solutions for RCM in GSD based on comparison and feedback results. Evaluation is performed by using a case study. The limitation of the framework is a lesser emphasis on the decision phase in RCM [25].

The author explains traceability benefits and flow in software development projects and proposes a traceability assessment model. The assessment model is used to determine the degree of traceability fulfilment. In previous literature, practitioners do not correctly guide how to achieve the existing traces designed and maintained for humans to avoid ambiguity. For this purpose, the authors researched on the elements of traceability designing and proposed a model that explains every element condition, traceability gate, and inappropriate aberration traceability problem [26].

This article explains how both states acceptable and unacceptable can be observed for their project traceability. The suggested framework is assessed using the supplied model and an expert survey, with participants concluding that it is effective, dependable, and compliant with the quality requirements. The limitation of this model is that it cannot be applied to identify semantically incorrect trace links [27].

The author explains in this article to trace the textual artefacts using traceability automated technique, and several measures are added to build a model that generates or recovers automated trace links. Logistic regression is a single-step model used to identify new trace links. This type of approach requires enough training in the required set of data, and the former is by far in the minority that may lessen the performance of such models [28]. Therefore, this article discusses the significant issues of identifying trace links as a challenge; for this purpose, the logistic regression model is presented that generates the trace links and finds out the accuracy rate of user requirements. To get the best possible logistic regression classifier, we must subtract links in the trace dataset that generates several issues and is not in the representative form [29].

The author describes the framework that enhances the effectiveness of RCM in GSD structured and semi-structured data collection techniques. From published papers, current case studies, and publications, data are extracted. The author presented an Online Remote Construction Management (ORCM) paradigm for ontological requirement change management. Data are extracted from published reports, existing case studies, and articles. The author proposed an ontological requirement change management (ORCM) model [30]. When a change request is imitated, the impact analysis of this change develops its UML and estimates its cost. After developing UML, it accepts or rejects change and switches to the ontology phase. Evaluation of this proposed model is performed by using experts' reviews. This framework improves the effectiveness of RCM in GSD. This framework guides the team of developers regarding how rapid changes could be consolidated [31].

The author found that project exploitation becomes more difficult as computer systems become more and more complex. Unfortunately, many system projects fail due to the lack of RM. The author proposed a Requisite-Pro (RM) tool in this article. This model establishes the UML and shows the consistency and traceability between requirements [32]. This framework contains qualities that reduce manual labor errors, is separated into lexicon and property, and provides a principle. Every criterion for this division is distinct from the others. The first need is to link with UML, followed by the phase of traceability. The outcome demonstrates that this method for managing modification requests is feasible and logical. For this division, every requirement is different from others. First, the requirement is to connect with UML and then go to the traceability phase. The result shows that this approach for RM handles all change requests, and it is practical and reasonable [33].

The author explains the requirement engineering concepts with development activities and defines how user requirements are converted into the model and programming language. The proposed framework is the requirement-oriented modeling and programming language (ROMPL) model, which starts with user requirements and metamodels. Then, define the language, take the example of the healthcare center to mention the requirements, and model the language. Requirements traceability is a challenging task, and the basic purpose for designing ROMPL is to mitigate the traceability issues [34]. The objective is to maintain the user requirement in textual and coding form by enabling software developers and business analysts. ROMPL maintains both textual and visual forms and sets the priority. According to the priority, it takes functional needs, and the limitation of ROMPL is that it is not present in the prototype interface; action semantics are assigned, but its usability and efficacy have not been evaluated [35].

The author proposed a requirement tracing tool for automated verification for front end users, validation, and analysis system (SAVVAS), the software front end processor that was based on an Ingres relational database. The language of this tool is Pascal, which permits requirement elicitation in the form of keywords and link words. These tools are enhanced and still independently used. According to the developer's perspective, several researchers like Abrahams and Barkley, Ramesh, Watkins, and Neal explain the value of requirements tracing and define the basic concepts such as forward tracing and backward tracing [36].

The author analyzed two different reference models for requirements traceability; these are low-level models of traceability and high-level model of traceability. Low-end users generate traceability links to check the requirement dependencies and type of requirements and how they are assigned to the system to validate these requirements, and divided into the correct form to stimulate change control [37].

The author presents an information retrieval (IR.) model, which is used to compute the query representation; it specifies the details of the document representation and the retrieval functionality. The basic IR model can be divided into three models, like the inference network, vector Boolean, and probabilistic model. The Boolean model has been named the RST model, the most criticized model used for IR. It is used for translating ambiguous queries. For example, the query terminal figure and documents are indexes with economic terms [38].

2 Materials and method

The research methodology allows the team member to assemble their work into an abstract and cohesive form, generate an idea for other task completion, implement the research plan in developing projects, and explain the flow of process like how data are gathered and analyzed. In software engineering, the researcher has a significant range of research methods that can be used for efficient and reliable research. Software engineering (SE) field identifies five phases of research methods that are most relevant, as shown in Figure 2.

2.1 Scope

This section defines the research objective, and research questions are formulated. It also explains how we use the methodology to produce a defined document by following the strict technical rules. The scope of our research and the methodology are as follows. This research focuses on RM activities, keeping the industry software development in view. The research methodology we have adopted is the case study. The reason behind the selection of this approach is to find out the gaps in previous research that we have made as base requirements. It provides an understanding of how problems were solved in the past, and we are in a better position to solve problems by using some techniques not used in previous research. The methodology which is used in this research work has the following steps. A literature survey has been done to diagnose the problem.

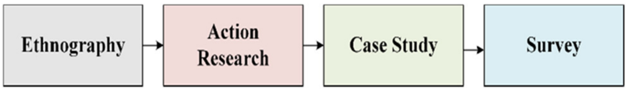


Figure 2: Research methodology.

1. Research several industries have different research methods (RM) scope and definition.
2. Depending on the demands of the developer, we are doing a personalized assessment of the research methods (RM) tools that specifies how to trace requirements using automated online tools.
3. To elaborate on the pros and cons of several tools.
4. Conduct a trade-off survey of several tools.
5. Utilizing case studies and industrial trials, the assessment has been conducted.
6. Give the direction for future research and improvements.
7. Describe the flow of research which is used in this article.

2.2 Action research

1. Problem identification.
2. Before starting action research, identify the area, and set the focus.
3. Discover the area of research that we want to explore, focus on the research area, which we want to improve or change?
4. Define the situation of change.
5. What is the additional change from previous research?
6. Why do we want to change?

2.3 Collect data

Data may be gathered from several sources. The research area discoversthe region's new dimensions. The following resources may be used to acquiredata:

1. Surveys
2. Observations
3. Journals
4. Portfolios
5. Interview

2.4 Analysis of data

Ensure the collected data fulfill the inclusion criteria and answer the questions satisfactorily. Analyze and organize the data in charts and graphs, explain the findings, and show the results.

2.5 Plan for future action

In the end, we have analyzed our work and gave answers to the following questions.

1. What have we achieved additionally in this research?
2. Is there work benefits for others?
3. Conclusion according to our expectations?
4. Can new questions be generated from this work?

2.6 Research questions

In this work, we originate the research questions to understand the RM better, as shown in Figure 3.

Its historical and current tools and approaches, as well as its merits anddownsides in a distributed setting. These research questions are as follow:

1. RQ1: What are the fundamental issues of requirement traceability in a distributed environment?
2. RQ2: What are the most challenging Research Methods (RM) concerns indispersed environments?
3. RQ3: How to handle the RM and traceability in a distributed environment in one frame?

2.7 Research string

A research string is used to refine the search. We use the keyword for the research string. We also use the synonyms of essential terms and logical operators to make the research string. The primary purpose of the research string is to provide the inclusion criteria papers that are directly linked to the topic. Combine the keywords with logical operators AND & OR to make the string.

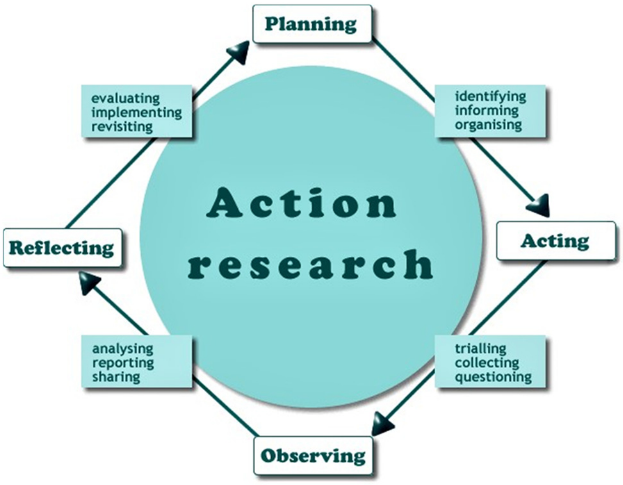


Figure 3: Action research.

Here is the list of research strings that we used for searching

1. Situational method engineering
2. Situational method engineering approaches
3. Requirement management in a distributed environment
4. Requirement traceability
5. Requirement management models and tools
6. Requirement traceability models
7. Action research
8. Global software development
9. Requirement change management in GSD
10. Requirement traceability in GSD
11. Traceability matrix (TM)
12. Requirements management tools
13. Requirement change management (RCM)
14. RCM AND GSD
15. RM OR traceability AND GSD
16. RM AND GSD
17. RM AND GSD OR RCM
18. Traceability AND GSD
19. RCM OR traceability AND GSD
20. RCM OR traceability OR RM
21. RCM OR traceability OR RM AND GSD

2.8 Perform research using inclusion and exclusion

We searched for papers based on the above strings and added them to our research. Many papers are available, but we excluded them if they did not fulfil the criteria. Only those papers are included which focus on RM, requirement traceability, RCM, and situational methods.

2.9 Proposed framework

RM is used to manage unclear and ambiguous requirements and makes it traceable whether they affect each other or not. The main aim of RM is to remove the ambiguity of requirements and generate the matrix of these requirements. The requirement specification phase also helps to make the base requirements clear by using case diagrams. The proposed framework is divided into three phases: pre-planning, development phase, and comparison phase, as shown in Figure 4.

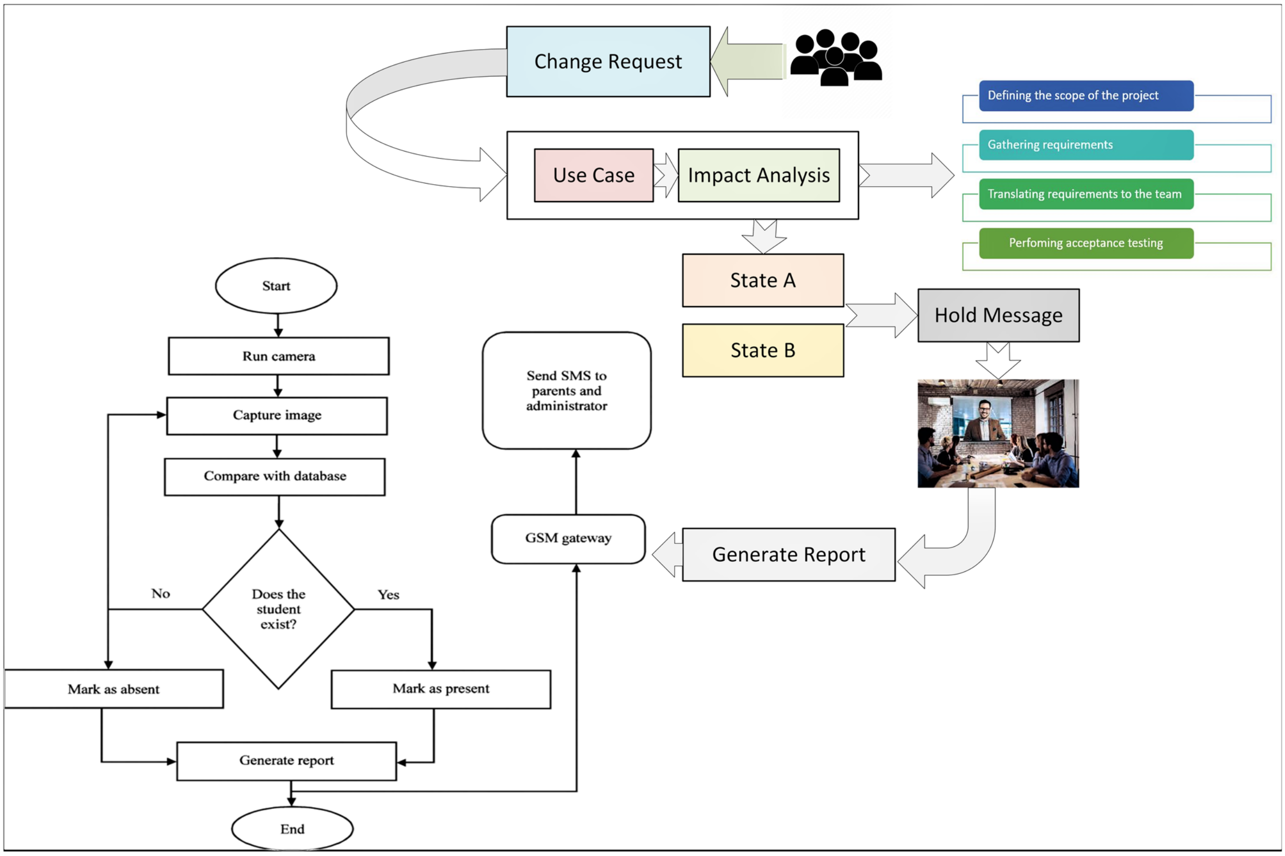


Figure 4: Proposed framework (FFRM).

2.10 Framework phases process flow

We divide the framework into three phases: pre-planning phase, development phase, and comparison phase.

2.10.1 Pre-planning phase

That is the initial phase. Requirements are specified in this phase. Requirement specification: The change request is analyzed when initiated in the pre-planning phase. Change is initiated according to the change in the environment. When a client requests for changes in any requirements, then this request is analyzed to see whether this change is possible or not. These analyzed requirements are modeled using case diagrams by requirements analysts to specify the flow of processes and conclude whether it has positive or negative effect on other requirements. In this section, user stories are written on index cards and these requirements are modeled to remove the ambiguity because these requirements are the project's base, so correction of base requirements is mandatory for successful project compilation. Impact analysis is done in this phase by requirement analysts. Responsibilities of requirement analysts are:

1. Gather user requirements
2. Plan and document
3. Test use cases
4. Analyze impact of change

These simplified requirements are sent to all dispersed sites, and employees start working on these requirements further. These validated requirements are sent to all sites to hold meetings using webcams. All the employees on both sites are available. Because of its dispersed nature, coordination issues occur, and webcam meetings resolve that. Suppose a change is initiated in-site A, it is mandatory to inform all other sites after validation of requirements change distributed to all sites, using webcams. If this process misses site A or B employees, they are stuck at the end of this phase as the report mismatches with user requirements. Finally, requirements are documented in the form of a report.

2.10.2 Development phase

This is the second phase in which requirements are checked by using the concept of traceability, constituting traceability events and TM.

2.10.2.1 Traceability links

In this step, requirements are checked by generating trace links.

2.10.2.2 Tracing the requirements

For the generation of traceability links, every requirement must be assigned with a unique label that removes the ambiguity between the requirements. Instead of writing the requirements in large paragraphs, write requirements in a fine-grained fashion. User requirements trace forward the requirements to show the effect of change during the software development, also to give the confidence that the requirements specification has fulfilled the user requirements, and we can trace the requirements from backwards to require identification. Traceability links give the idea of dependent requirements and explain the effect on other dependent requirements when any requirement is changed or deleted. Test cases are generated after requirement tracing that identifies the unimplemented requirements. For example, tracing and testing the requirements for verification purposes help us identify the test case that needs to be updated after accepting any change. Maintaining traceability is the greatest challenge because the tracing of requirements continues until the end of development. Previous literature shows that change cost contributes to 40–90% of the total development cost.

2.10.2.3 Traceability events

Traceability is a process in which all the requirements are linked to each other and communicated until the end of the validation process. Traceability transforms a high-level language (requirements, purpose, and objective) into a low-level language. It maps and traces the user requirements generating test cases. They are traceability events as follows:

1. Decompose
2. Generate links
3. Notify

2.10.2.3.1 Decompose

In this phase, requirements are divided to check traces and remove ambiguity in requirements. Because if requirements are written in a complex form like large paragraphs that might create ambiguity for the developer, so it is necessary to decompose the user requirements for further implementation.

2.10.2.3.2 Generate links

After decomposition checks the dependencies of requirements, we analyze how these requirements are affected by each other if any change occurs. It generates links between requirements for checking their dependencies.

2.10.2.4 TM

A TM is a spreadsheet or table that presents the test cases regarding each requirement. The primary purpose of maintaining the TM is to check whether the requirements are tested after a change. It also checks whether the user requirements are fulfilled. The TM of SE supports the change impact analysis, regression testing, verification, and validation. TM plays a prime role that shows the successful implementation of requirements. Nowadays, many researchers are applying the IR approach for tracing requirements. In this technique, traces are automatically generated.

2.11 Parameters for TM

1. Requirement type and explanation
2. Requirement Id
3. Unit test cases
4. Challenges and issues

2.12 IR approach

In traceability generation, two IR models are used: vector space model and probabilistic model. The IR approach generates the trace links and also checks the requirement dependencies on each other, how much they are dependent, and also checks whether they are according to the given requirements which are written on an index card; if these are not according to the need, then turn back to previous steps and again check the dependencies and trace links otherwise send to the data repository for data management, and adaptation process also notifies the sites A and B. For updating and maintaining traceability relationships, event-based approach is used, as shown in Table 1.

2.13 Traceability relationship

In traceability relationships, dependent requirements like how they depend are mentioned. The relationship between the requirements is explained. In case of any change, dependent requirements are notified to adopt proper operations. This method has two main components, i.e., the requirement manager manages the requirements for event messages to the event server and publishing change. The subscriber manager is responsible for listening to the subscribers it manages for event notifications forwarded by the event server. Before event-based algorithms, EB assumes that the traceability links among artefacts have been established. Now, the focus of the algorithms is to manage the traces according to the suggested change.

2.14 Boolean model

This model is used where requirements depend on each other. Changes in any requirements can have a direct effect on other requirements. Suppose the user asks for the change in two requirements, if requirement R1 is added, then R2 is needed to implement R1.

2.14.1 Comparison phase

In this phase, the output of the upper module and the report generated at the end of the pre-planning phase are compared to check that both the requirements are traced, and matrices are generated according to the finalized report and to check whether the system is user-friendly and cost-effective. If the results of these two modules are exactly saved in the data repository after saving, the report is generated again. Now we have two reports, one after specification of requirements and the second after saving in the repository. The two reports can be compared to see what change is requested and what we have achieved.

3 Results and discussion

3.1 Case I

A case study is an event and activity that have an ideal situation with some real-life complexities that we want to mitigate. It also explains how complexities affect real-life decisions. For explaining and evaluating our recommended framework, scenarios are formulated and evaluated. A case study is to apply our knowledge and skills in real-world situations. Some steps or processes are followed to apply in the case study.

1. First analyze
2. Apply knowledge
3. Give reasoning
4. Composition of conclusion

3.1.1 Feature of case study

Here are some features of a good case study:

1. Taken from real-life examples
2. Consist of many phases that end with problem ambiguity of requirements
3. It gives enough information to the reader to handle the problem

Table 1: Requirement dependencies

Interdependency type	R1 requires R2	R1 refines R2	R1 conflicts with R2
R1 is modified	R2 is not affected	R2 is not affected	R2 is not affected
R2 is modified	R1 is candidate affected	R1 is candidate affected	R1 is not affected
R1 is deleted	R2 is not affected	R2 is not affected	R2 is not affected
R2 is deleted	R1 is affected	R1 is candidate affected	R1 is not affected
New R is added to R1	R1 is affected, and R2 is not affected	R1 is affected, and R2 is not affected	R1 is affected, and R2 is not affected
New R is added to R2	R1 is candidate affected, and R2 is affected	R1 is candidate affected, and R2 is affected	R1 is not affected, and R2 is affected

3.1.2 Framework evaluation

Moftak solution is a software company that provides software solutions around the globe. Moftak started his journey in 2008 with a small structure. They have 50–100 employees in countries like Pakistan, the United Kingdom, Canada, and Australia. Also, they have a great collaboration between all these branches. They have a dazzling and qualified expert group that makes the clients happy through their shining skills. They also adopt the latest tools and technologies to keep their employees updated every time.

3.1.3 Expertise

They can handle any project, from static to dynamic solutions, their latest products are:

1. Wheels cab service
2. TV websites
3. Lab management system
4. Delegation management system
5. Medical health
6. Inventory control system

The main area of interest is developing Islamic websites with online Islamic channels, videos on demands, an online Quran learning platform, a charity donation system, and much more related to Islamic points of view.

Step 1

In the 1st step, the relevance criteria of the framework and case study are explained and why this case study was picked for this framework.

Step 2

The second step explains the difference between the given case study and the framework.

Step 3

In the third step, we test some hypotheses for this research.

Hypothesis

In this section, we test some hypotheses by assuming to fulfil the final criteria that make it better than others. The FFRM is different from other frameworks because in FFRM, in the first phase, requirement analysts specified the change requirements by using use cases that give their assumption related to change. Assumptions are given below the time consumption and estimated budget.

Step 4

In this step, we explain the effect of change.

3.1.4 Ripple effects of change

In a dynamic environment, change requirements affect the traceability matrices in which previous requirement dependencies are mentioned against every requirement. Suppose R1 and R2 are two requirements. Specified requirements were sent to all other sites using online web tools to solve the communication issue. At the end of this phase, a report is generated for further analysis; requirements are managed as shown in Tables 2 and 3.

3.2 Case II

We have selected wheels' cab service as case II. The primary purpose of wheels is to provide a distributed system to enhance the public interest. It is a web-based system with mobile applications to keep our clients more flexible. Wheels connect more people and cities with a simple, technology-enabled, real-time platform for safer and more accessible cab service. It plays an essential role in user gratification with our best way out.

3.3 Expert review

A case study which we used in our study as explained above. We evaluate our framework through the case study, conduct an online crust survey, and evaluate through the tool; based on the expert's review, we prepare a percentage graph as shown in Figure 5.

Table 2: Relevance criteria

FFRM	Case study
Tackle requirement traceability	Improve requirement traceability
Tackle change impact in GSD	Manage change
Check requirement dependences	Check requirement dependences

Table 3: Difference

FFRM	Case study
Data repository	Use agent base ontology
Maintain auto TM	Multiple agents perform the task

3.4 Comparison with existing tools

We compare our framework with previous tools and techniques by using some parameters. These parameters are described in Table 4.

3.5 Industrial review

We conducted an online crest survey to get reviews of experts about FFRM. We prepared the questionnaire to check the reliability and effectiveness of our framework. The online link of the crest survey was also sent to the industrial experts and our colleges. All of them gave feedback. Based on these responses, we generated the automated graph in crest software.

3.6 Representation of response

Response given by the experts are shown in a graph.

3.7 Industrial description

For the evaluation of our work, we conducted an industrial survey. The survey was conducted in the following two companies, Eziline software house and Moftak solutions. The number of experts in this company was 10–100. All

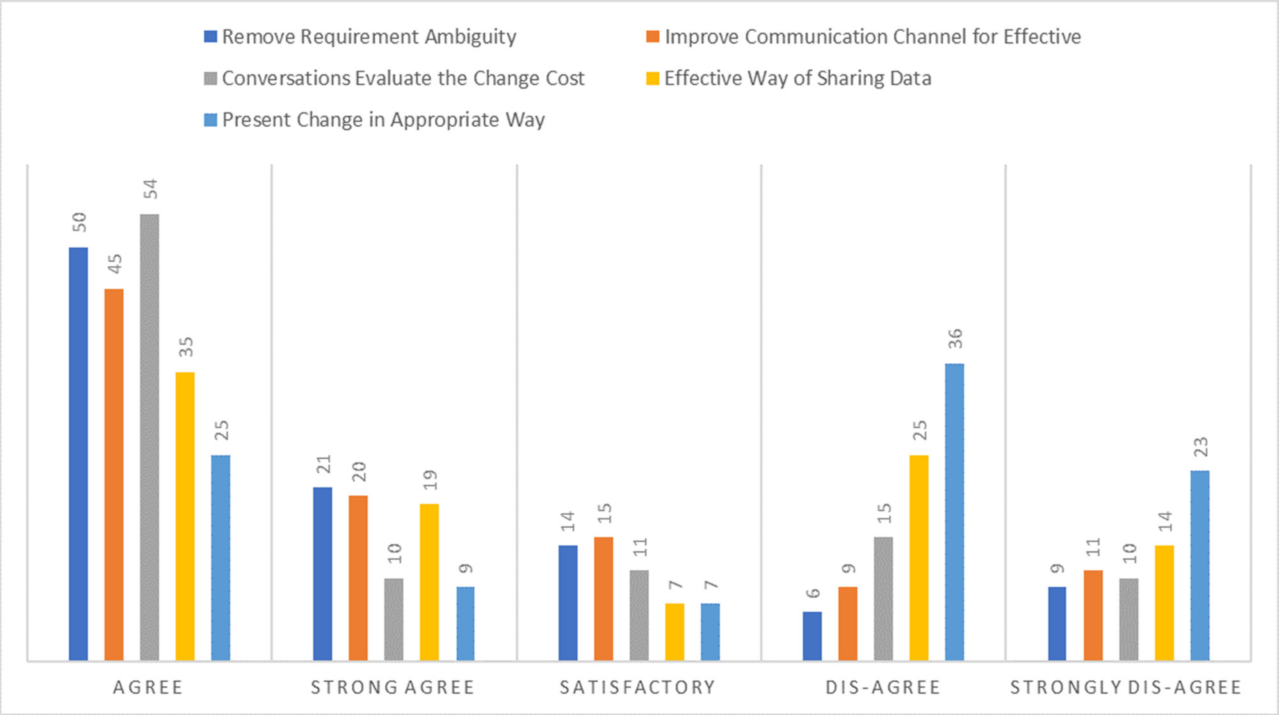


Figure 5: Expert review.

Table 4: Comparison of framework with previous tools and techniques

	HYCAT	TAM	Multi-agent architecture	FFRM	ROMPL	Requisite pro	ORCM
Change request	✓	✓	✓	✓	✓	✓	✓
Use cases	✗	✗	✗	✓	✓	✓	✓
Impact analysis	✓	✗	✗	✓	✓	✗	✓
Generate report	✗	✗	✗	✓	✗	✗	✗
Traceability events	✓	✓	✓	✓	✓	✗	✗
Traceability with IR and Boolean model	✓	✓	✓	✓	✓	✗	
Comparison	✗	✓	✗	✓	✗	✗	✓
Data repository	✓	✓	✓	✓	✓	✗	✗
Report generation	✗	✗	✗	✓	✗	✗	✗

these experts had considerable working experience in this field.

3.8 The evaluated results of survey questions

The survey question (Sq.) is designed in crest software to evaluate the proposed framework by extracting the feedback from Eziline and Moftak solutions. We created nine research questions for our framework evaluation. The representation of data is based on the feedback of experts. These survey questions are listed below, and their graphical visualization is shown in Figure 6.

Sq.1 Use of RM in a distributed environment? Feedback of this question is that 67% of people agreed, 6% disagreed, 8% neutral, and 19% were satisfied using RCM.

Sq.2 Managing traceability using RM and RCM? Feedback of this question is that 67% of people agreed, 6% disagreed, 15% neutral, and 12% satisfactory.

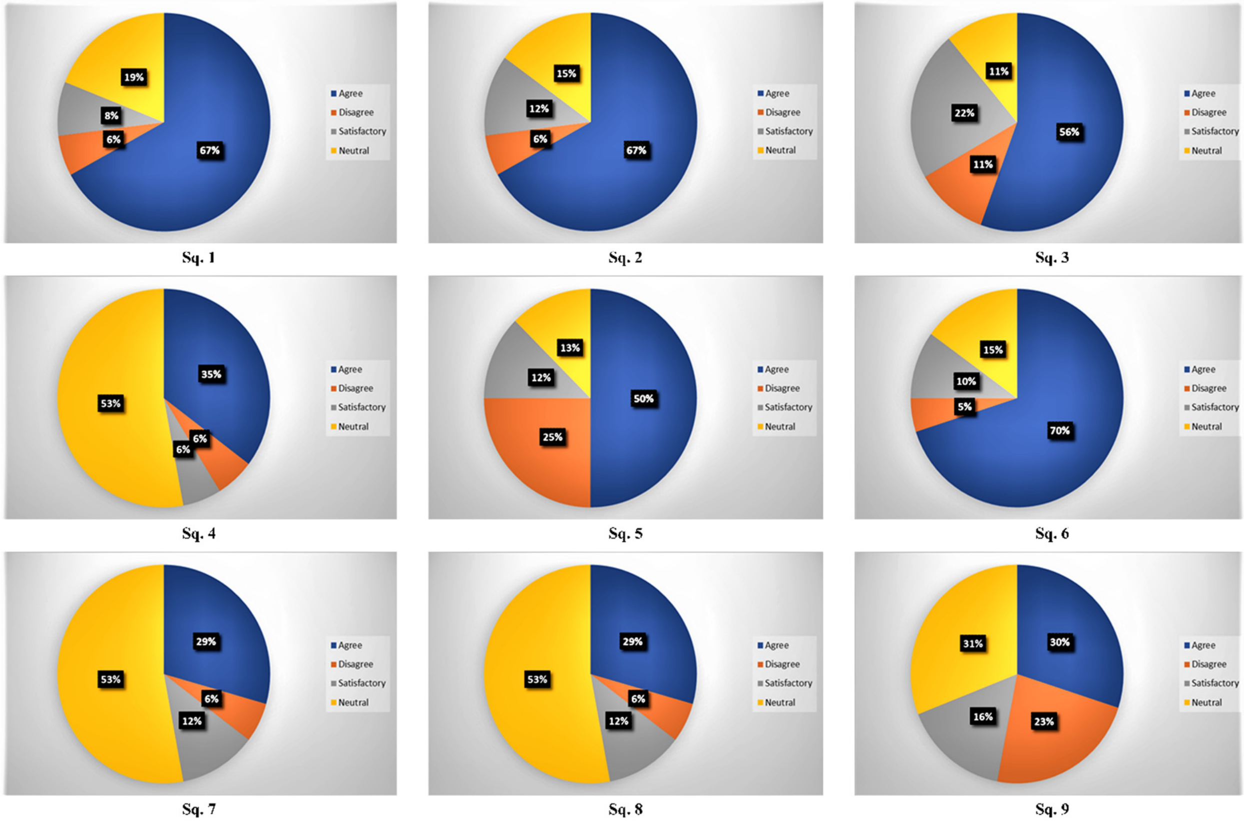
Sq.3 Is a document useless if not traceable and managed? The feedback of this question is that 30% of people agreed, 23% disagreed, and 31% neutral, 16% were satisfactory.

Sq.4 Does the framework give the expected output? Feedback of this question is that 50% of people agreed, 18% disagreed, 10% neutral, and 22% are satisfied that this framework gives the expected output.

Sq.5 Should we agree to implement this framework in the software industry? Feedback of this question is that 68% of people agreed, 10% disagreed, 9% neutral, and 13% were satisfied by implementing this framework in the software industry.

Sq.6 Is this system flexible in small and large organizations? Feedback of this question is that 45% of people agreed, 25% disagreed, 14% neutral, and 16% were satisfied using this framework in all (small or large) organizations.

Sq.7 The need for traceability and management is compulsory for valuable documents? Feedback of this question is that 70% of people agreed, 5% disagreed, 10% neutral, and 15% are satisfied that a document is needed for RM and traceability.



Sq. 1

Sq. 2

Sq. 3

Sq. 4

Sq. 5

Sq. 6

Sq. 7

Sq. 8

Sq. 9

Figure 6: The output of Sq.1 to Sq.9.

Sq.8 Can this framework be used to overcome the management issues? Feedback of this question is that 54% of people agreed, 12% disagreed, 9% neutral, and 25%satisfiedy using FFRM to mitigate the RM issues.

Sq.9 Give the percentage/% if these modules are in flow? Feedback of this question is that 44% of people agreed, 12% disagreed, 15% neutral, and 29 % are satisfied with the flow of these modules. Percentage of RM in a distributed environment.

4 Conclusion

The primary purpose of this research was to improve the requirement traceability by creating traceability links in a distributed environment. Our framework consists of three phases. The first phase is about removing the requirement ambiguity and generating the report. The second phase makes the requirement traceable. Generate the traceability links and check the requirement dependencies. Use an IR approach for traceability and check whether the requirements are according to user needs. If valid, save it in the database repository.

This research has explained the importance of RM and requirements traceability. Requirements traceability is generally considered a time-consuming and expensive task in a distributed environment, and it becomes more complex in distributed sites because people work from different locations and have coordination and time problems. To overcome this issue, an automated system that manages the change request, checks the dependencies of requirements to each other, and generates its test cases to check the requirements as the customer wants is proposed. Most of the projects fail because of their mismanagement in requirement ambiguity. We proposed an FFRM to mitigate these issues. The framework is evaluated through a pilot case study and an online survey is conducted. The experts feedback are displayed in the form of a graph which shows the percentage of agree, disagree, satisfactory, and neutral. We have asked whether implementing this framework in the industry is reliable, and most of the experts suggest applying this framework in the software industry.

4.1 Future work

The primary objective of this research is for the RM and a framework is proposed that solves the RM issues. However, RM is a vast field, and there is still scope for further research, particularly by the adaption of new RM and traceability tools.

Funding information: Authors state no funding involved.

Conflict of interest: Authors state no conflict of interest.

Reference

[1] M. Humayun and N. Z. Jhanjhi, “Exploring the relationship between GSD, knowledge management, trust and collaboration,” *J. Eng. Sci. Technol.*, vol. 14, no. 2, pp. 820–843, 2019.

[2] S. S. A. Bukhari, M. Humayun, S. A. A. Shah, and N. Z. Jhanjhi, “Improving requirement engineering process for web application development,” *In 2018 12th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics* (*MACS*), IEEE, 2018, pp. 1–5.

[3] S. Wang, J. Wan, D. Zhang, D. Li, and C. Zhang, “Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination,” *Comput. Netw.*, vol. 101, pp. 158–168, 2016.

[4] A. Vargas, L. Cuenca, A. Boza, I. Sacala, and M. Moisescu, “Towards the development of the framework for inter sensing enterprise architecture,” *J. Intell. Manuf.*, vol. 27, no. 1, pp. 55–72, 2016.

[5] M. Qasim, A. Bibi, S. J. Hussain, N. Z. Jhanjhi, M. Humayun, and N. U. Sama, *Test case prioritization techniques in software regression testing: An overview*, *International Journal of advanced and applied sciences*., vol. 8, no. 5, pp. 107–121, 2021.

[6] D. Sabella, A. Vaillant, P. Kuure, U. Rauschenbach, and F. Giust, “Mobile-edge computing architecture: The role of MEC in the Internet of Things,” *IEEE Consum. Electron. Mag.*, vol. 5, no. 4, pp. 84–91, 2016.

[7] O. Turetken, I. Stojanov, and J. J. Trienekens, “Assessing the adoption level of scaled agile development: a maturity model for Scaled Agile Framework,” *J. Software Evol. Process.*, vol. 29, no. 6. p. e1796, 2017.

[8] A. I. Böhmer, M. Meinzinger, R. Hostettler, A. Knoll, U. Lindemann, “Towards a framework for agile development of physical products influence of artifacts and methods,” in 2017 International Conference on Engineering, Technology and Innovation (ICE/ITMC), IEEE, 2017, pp. 237–245.

[9] M. A. Moisescu and I. S. Sacala, “Towards the development of interoperable sensing systems for the future enterprise,” *J. Intell. Manuf.*, vol. 27, no. 1, pp. 33–54, 2016.

[10] R. Guerzoni, I. Vaishnavi, D. Perez Caparros, A. Galis, F. Tusa, P. Monti, et al., “Analysis of end-to-end multi-domain management and orchestration frameworks for software defined infrastructures: an architectural survey,” *Trans. Emerg. Telecommun. Technol.*, vol. 28, no. 4. p. e3103, 2017.

[11] J. Bogner and A. Zimmermann, “Towards integrating microservices with adaptable enterprise architecture,” *In 2016 IEEE 20th International Enterprise Distributed Object Computing Workshop* (*EDOCW*), IEEE, 2016, pp. 1–6.

[12] P. Pelliccione, E. Knauss, R. Heldal, S. M. Ågren, P. Mallozzi, A. Alminger, et al., "Automotive architecture framework: The experience of volvo cars," *J. Syst. Archit.*, vol. 77, pp. 83–100, 2017.

[13] S. Tyagi, R. Sibal, and B. Suri, "Empirically developed framework for building trust in distributed agile teams," *Inf. Softw. Technol.*, vol. 145, p. 106828, 2022.

[14] N. A. Santos, J. Pereira, N. Ferreira, and R. J. Machado, "Using Logical Architecture Models for Inter-Team Management of Distributed Agile Teams," *Int. J. Inf. Technol. Syst. Approach* (*IJITSA*), vol. 15, no. 1, pp. 1–17, 2022.

[15] Y. I. Alzoubi and A. Q. Gill, "Can Agile Enterprise Architecture be Implemented Successfully in Distributed Agile Development? Empirical Findings," *Glob. J. Flex. Syst. Manag.*, vol. 23, pp. 315–330, 2022.

[16] C. Nundlall and S. D. Nagowah, "Task allocation and coordination in distributed agile software development: a systematic review," *Int. J. Inf. Technol.*, vol. 13, no. 1, pp. 321–330, 2021.

[17] H. Muccini, K. Vaidhyanathan, "Software architecture for ML-based systems: what exists and what lies ahead," *In 2021 IEEE/ACM 1st Workshop on AI Engineering-Software Engineering for AI* (*WAIN*), IEEE, 2021, pp. 121–128.

[18] M. Campo, A. Amandi, and J. C. Biset, "A software architecture perspective about Moodle flexibility for supporting empirical research of teaching theories," *Educ. Inf. Technol.*, vol. 26, no. 1, pp. 817–842, 2021.

[19] A. Hübner, H. Kuhn, and J. Wollenburg, "Last mile fulfilment and distribution in omni-channel grocery retailing: a strategic planning framework," *Int. J. Retail. Distrib. Manag.*, vol. 44, no. 3, pp. 228–247, 2016.

[20] S. Ghanadbashi and R. Ramsin, "Towards a method engineering approach for business process reengineering," *IET Softw.*, vol. 10, no. 2, pp. 27–44, 2016.

[21] S. Bondar, J. C. Hsu, A. Pfouga, and J. Stjepandić, "Agile digital transformation of System-of-Systems architecture models using Zachman framework," *J. Ind. Inf. Integr.*, vol. 7, pp. 33–43, 2017.

[22] A. Brown, J. Fishenden, M. Thompson, and W. Venters, "Appraising the impact and role of platform models and Government as a Platform (GaaP) in UK Government public service reform: Towards a Platform Assessment Framework (PAF)," *Gov. Inf. Q.*, vol. 34, no. 2, pp. 167–182, 2017.

[23] D. Larson and V. Chang, "A review and future direction of agile, business intelligence, analytics and data science," *Int. J. Inf. Manag.*, vol. 36, no. 5, pp. 700–710, 2016.

[24] Y. Masuda, S. Shirasaka, S. Yamamoto, and T. Hardjono, "An adaptive enterprise architecture framework and implementation: Towards global enterprises in the era of cloud/mobile IT/digital IT," *Int. J. Enterp. Inf. Syst.* (*IJEIS*), vol. 13, no. 3, pp. 1–22, 2017.

[25] C. Yang, P. Liang, and P. Avgeriou, "A systematic mapping study on the combination of software architecture and agile development," *J. Syst. Softw.*, vol. 111, pp. 157–184, 2016.

[26] S. Abdelkebir, Y. Maleh, and M. Belaissaoui, "An Agile Framework for ITS Management In Organizations: A Case Study Based on DevOps," *in* Proceedings of the 2nd International Conference on Computing and Wireless Communication Systems, ACM, 2017, p. 67.

[27] P. Jiang, K. Ding, and J. Leng, "Towards a cyber-physical-social-connected and service-oriented manufacturing paradigm: Social Manufacturing," *Manuf. Lett.*, vol. 7, pp. 15–21, 2016.

[28] A. Martini, and J. Bosch, "A multiple case study of continuous architecting in large agile companies: current gaps and the CAFFEA framework," *in 2016 13th Working IEEE/IFIP Conference on Software Architecture* (*WICSA*), IEEE, 2016, pp. 1–10.

[29] W. Song, "Requirement management for product-service systems: Status review and future trends," *Comput. Ind.*, vol. 85, pp. 11–22, 2017.

[30] N. Z. Bawany and J. A. Shamsi, "SEAL: SDN based secure and agile framework for protecting smart city applications from DDoS attacks," *J. Netw. Comput. Appl.*, vol. 145, p. 102381, 2019.

[31] M. Bashari, E. Bagheri, and W. Du, "Dynamic software product line engineering: a reference framework," *Int. J. Softw. Eng. Knowl. Eng*, vol. 27, no. 2, pp. 191–234, 2017.

[32] P. D. Ciampa, E. Moerland, D. Seider, E. Baalbergen, R. Lombardi, R. D'Ippolito, "A collaborative architecture supporting AGILE design of complex aeronautics products," *in 18th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2017, p. 4138.

[33] E. Rauch, P. Dallasega, and D. T. Matt, "Distributed manufacturing network models of smart and agile mini-factories," *Int. J. Agile Syst. Manag.*, vol. 10, no. 3–4, pp. 185–205, 2017.

[34] A. Sever, "Modeling Distributed Agile Software Development Utilizing Cloud Computing: A Holistic Framework," *Curr. J. Appl. Sci. Technol.*, pp. 1–12, 2019.

[35] P. Srivastava and S. Jain, "A leadership framework for distributed self-organized scrum teams," *Team Perform. Manage. An. Int. J*, vol. 23, no. 5/6, pp. 293–314, 2017.

[36] P. Abrahamsson, O. Salo, J. Ronkainen, and J. Warsta, "Agile software development methods: Review and analysis," *arXiv preprint arXiv:1709.08439*, 2017.

[37] S. Erol and W. Sihn, "Intelligent production planning and control in the cloud–towards a scalable software architecture," *Procedia CIRP*, vol. 62, pp. 571–576, 2017.

[38] A. Elzamly, B. Hussin, S. Abu Naser, K. Khanfar, M. Doheir, A. Selamat, et al., "A new conceptual framework modelling for cloud computing risk management in banking organizations," *Int. J. Grid Distrib. Comput.*, vol. 9, no. 9, pp. 137–154, 2016.