A Distributed Software Project Management Framework

Kamal Uddin Sarker [1],*, Raza Hasan [2],*, Aziz Bin Deraman [1], and Salman Mahmmod [3]

[1] Department of Informatics, Fakulti Teknologi Kejuruteraan Kelautan dan Informatik, Universiti Malaysia Terengganu, Terengganu, Malaysia; Email: a.d@umt.edu.my (A.B.D.)
[2] Department of Computing and IT, Global College of Engineering and Technology, Muscat, Oman
[3] Department of IT, School of Science and Engineering, Malaysia University of Science and Technology, Selangor, Malaysia; Email: salman.mahmood@phd.must.edu.my (S.M.)
*Correspondence: kusarker@umt.edu.my (K.U.S.), raza.h@gcet.edu.om (R.H.)

Abstract—**The software industry is enjoying the permeable and trans-border flow of software markets and can access resources from all corners of the world. Software engineers gain international work experience through a distributed working environment. It involves participation from individuals with different cultures, languages, and geographic time zones to work on a single project. In addition to providing global opportunities for software experts and businessmen, it also introduces new project management challenges. Barriers exist in trust, communication, monitoring, languages, cultures, and time zones. Distance mode management faces more technical challenges due to stakeholders' ambiguous understanding and various documentation. This study addresses an in-depth analysis of challenges and currently practicing methods. Moreover, a new virtual project management framework is proposed to minimize issues and maximize the virtual project management team's throughput. The framework is compared with commonly used methodologies by experts who have experience in global software project management, and the analysis is performed using the analytical hierarchy process. The evaluation matrix has shown that the proposed model is adequate for distance project management with better score in virtual scope and virtual management). Its excellency is in standard documentation practice, change management, and improving re-usability practice that will enhance business goals and stakeholder's satisfaction.**

Keywords—**virtual software project, distibuted project management, methodology, analytical hierarchy process, sustainability, business goals**

I. Introduction

According to the Harvard Business Review early in this decade, one-sixth of projects went over budget by 200% and an average 27% overrun on an intended budget, with a 5–15% failure rate [1]. According to the recent report of Project Management Institute (PMI) [2], 80% of projects can complete on time without significant extension of budget but poor performance has been noted for efficiency (high time complexity) and portability (minimum portability support). Standish Group is a well-known organization that publishes a report every year called "CHAOS report". It consists of root-cause analysis on the reasons for software project failure and statistical analysis for success rate ("projects are completed on time and budget", "failure rate"). Projects are stopped without submission, and challenging rate ("Projects are submitted but not fulfilled all requirements". "CHAOS report of 2018" noticed that approximately 30% project failed, 20% passed, and 50% challenged from the last five years [3]. According to the Geneca report [4], 75% of opinions from IT executives or business executives is that their project doomed from the start. This research addresses the significance of software project failure rate and shows the importance to improve process management activities of a software project through a methodology. PMI found that project management approach can support to mamage a software development processes effectively that could reduce additional costs and improve the reliability of a product [2]. Timo [5] showed a failed project does not only depend on an individual reason or action; but there is a causal relationship among people, processes, tasks, and the environment. This relationship directly indicates the performance of stakeholders and especially the role of management who are managing, coordinating, communicating, and keeping control of the project. Twelve activities (poor management and commercial pressure, stakeholder politics, unarticulated project goals, estimation error, inaccurate requirements specification, weak reporting system, sloppy practice in development, not capable to manage project's complexity, immature technology selection, unmanaged risk, and informal or poor communications) of projects are blamed to make a project failure by Charette [6], which are included in the software project manager's common activity list. Charette also emphasizes the importance of attaining complete knowledge of a project before commencing. The Project Management Office (PMO) is a white paper that is published by PMI [7] to support strategic level decision making by using tools, methodology, and policy-procedure in a project. It suggests to include managerial

Manuscript received November 19, 2022; revised February 23 2023; accepted April 18, 2023; published July 19, 2023.

doi: 10.12720/jait.14.4.685-693

activities to a software development life cycle process to accomplish a project successfully. Managerial activities such as work distribution, monitoring, tracking, work integrating, and communication become more complicated when stakeholders are connected through a virtual environment from a different physical location (i.e., country or state), language, culture, and time zone.

Workplace is an online working environment where stakeholders have no physical interaction. Project manager distribute the works to the stakeholders (employee, customers, suppliers, or anybody else who has responsibility in this project) who are in the different time zone (physical location). This also can be called distributed project management. This kind of work helps to achieve business goals by reducing manpower and logistic cost of a project though it has managerial challenges [8, 9].

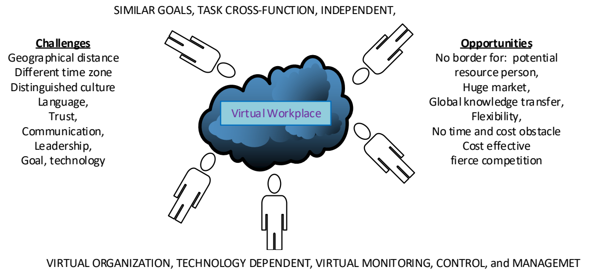

SIMILAR GOALS, TASK CROSS-FUNCTION, INDEPENDENT,
Challenges
Geographical distance
Different time zone
Distinguished culture
Language,
Trust,
Communication,
Leadership,
Goal, technology
Virtual Workplace
Opportunities
No border for: potential resource person,
Huge market,
Global knowledge transfer,
Flexibility,
No time and cost obstacle
Cost effective
fierce competition
VIRTUAL ORGANIZATION, TECHNOLOGY DEPENDENT, VIRTUAL MONITORING, CONTROL, and MANAGEMET

Figure 1. Understanding virtualization project management environment, challenges, and opportunities.

Fig. 1 illustrates the virtual workplace for software firms that consists of list of opportunities and challenges to the right most and left most side respectively. A virtual environment is technology dependent and an organization support with infrastructure so that project's functionality can monitored, manage, and control. Independent problem solving skill is one of the most important qualities are required to work in this mode of service. Though the team members are independent but they are in a group with similar goal and have cross function task/job/responsibility. The manager of a distributed project divides the task among team members and they interact with a virtual project management application for collaboration, sharing, communication, and to accept/give feedback.

Virtual team members are located in different geographical locations and are distinguished by factors such as culture, language, and time zones, as shown in Fig. 1. It is difficult to build trust, responsiveness, and good relationship among the stakeholders due to the cultural barriers, different time zone, and geographical distance. These challenges may also lead to project delays or failures [10]. Non-verbal communication of the team members also create a threat to accomplish a project effectively [11]. The project management institute's report [12] mentions eight factors (task complexity, communication technology, team membership, social, political, cultural, temporal, and geographical dispersion) these become more challenging in a virtual workplace.

This research proposed and evaluated a new framework for software project management that focuses on virtual project management activities. The objective of this methodology is to minimize the challenges of virtual software project management. Rest of the paper is organized as follows: Section II consists of literature review and the proposed model is presented in Section III. Evaluation criteria are analyzed in the Section IV and Section V performed an evaluation by a set of experts based on the selected criteria. Concluding remarks are given in Section VI with limitation and future work.

II. Literature Review

To improve the success rate, sustainability, and business goals of a software firm, outsourcing is often used. It allows for the utilization of comparatively cheaper talent from around the world, who can enhance business goals and expand the scope for talent to show competence in the global market. However, hiring the appropriate talent and effectively monitoring and controlling outsourced projects can be challenging. When a projects are distributed across multiple physical locations, communication problems due to cultural, linguistic, and timezone differences can arise. This paper proposes a framework to overcome these challenges and enhance managerial efficiency, as depicted in Fig. 2. This section emphasizes the importance of having a clear methodology or framework, highlights current issues in software projects, and discusses methods for evaluating the proposed framework.

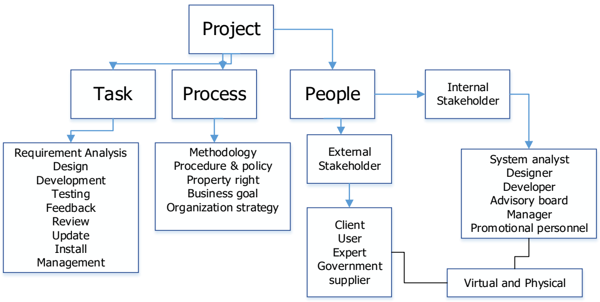


Figure 2. Virtualization scope in distributed software project management.

A. *Global Issues in Virtualization Project Management*

A software project is a subassembly of an information technology project that performs explicitly application development. An application is a soft product that has scope to develop from anywhere from the world. Organizations also utilize comparatively low labor, though selecting and managing an appropriate person is difficult. The focus of this work is on managerial activities. A resource person who works from the outside of the organization is an internal and an external stakeholders (Fig. 2). The study of Abdulrazak [13] underscored cultural issues, political factors, legal factors, and economic factors are common challenges of an international project. Additionally, it recommends the importance of multiple strategies for generic language, different time zone and sociability, and bridging cultural barriers of international organizations. Technical and communication conflicts arise due to cultural diversity, communication skill gaps, and ambiguous terminologies. Our work proposed a methodology that minimizes those challenges. It emphasizes the importance of developing a sophisticated common working environment to overcome these challenges. Additionally, technical projects may face difficulties in delivering products and sharing knowledge when the backlog is full and capacity is limited. Furthermore, international projects are subject to greater complexity and enforcement challenges. "The state of the software development" [14] statistics have shown that the most priority challenge is managing capacity and methodology implementation. The nearest second issue is knowledge sharing and regular meetings is suggested. Hiring talent is ranked third according to the developer's view. Moreover, CodingSans [14] reports that scrum is in the highest pick with 60% user's choice, while Kanban is 35%, agile modeling is 19%, and so on. 75.83% of respondents answered "yes" when asked if they use testing tools, with "Jenkins" and "Selenium" being the most popular choices. 57.70% uses Jira, 34.53% uses GitHub, 19.86% uses BitBucket for project management according to the choice rate of tool selection. Testing is crucial in virtual software projects [9, 15], and it can be minimized through automated testing. Nature-inspired algorithms [16], fuzzy rules [17], and automated test case generation techniques [18] are some of the recent approaches for reducing the test suite in virtual software projects. In a distributed project, a method of selecting priority nodes can be applied to reduce test cases, which improves the effectiveness of automated testing [18, 19].

B. Role of Methodology

The historical development of the methodology aims to reduce risk, avoid duplication of effort, and provides effective management. A methodology helps to achieve business goals, improves formal communication standardize documentation, increase the reusability of design and code, helps to make a successful project management team. Software development is continuously upgrading by the utilization of newly invented technology and applications. For example, in the 2018 blockchain, the Internet of Things (IoT) and Artificial Intelligence (AI) was the trending technology in the software industries while 2019 has come up with much better uses of this technology [20]. Technological changes introduce new challenges in software management, for example, the state forward blockchain technology issues [21]. The IoT project is commonly a multi-disciplinary platform and creates demand for a comprehensive representation of stakeholders as well as more importance is given to recording information about IoT objects [22]. AI software projects are complicated with requirement specification, relationship generating, and functional dependency that brings challenges for software engineering [23]. The methodology also needs to update by addressing the upcoming technological advancement to utilize the opportunity and resolving challenges. Our proposed methodology is addressing created issues through technological advancement.

C. Framework Evaluation Techniques

The Multi-Criteria Decision-Making (MCDM) or Multi-Criteria Decision-Analysis (MCDA) are being executed in the operational research to explicitly assess conflicting criteria. Service cost and customer satisfaction are conflicting factors in the service industries [24]. MCDM has been proposed for providing ranking and taking a decision of optimal point selection [25] the objects are featured or characterized [26], a method selection for identification of the appropriate application [27], methodology evaluation for sustainability practice [28] and so on. This technique is adapted to the expert system by a fuzzy approach [29] and is updated by a criteria-based approach [30]. Analytical Hierarchy Process (AHP) is one of the MCDM that is used to compare the proposed model with similar existing models.

D. Related Work

Software methodologies have been developed since the beginning of the software industry. The Waterfall model, which is a pioneer in plan-driven approaches, was invented before the concept of virtual project management [31] and has been modified in subsequent versions [32]. The incremental approach is implemented in different Agile methodologies, but this scheme focuses on partial solutions, making it more suitable for informal approaches and not justified for virtual project management [33]. The Spiral model is specialized for high-risk models and not highlighted for distributed projects [34]. The V-model is more concentrated on the quality of the product and does not consider distributed project management [35]. PRINCE-2 is a generalized model for any kind of project management that concentrates on virtual project management but does not focus on software projects [36]. This research proposes a framework only for the software industry.

Sarkert *et al.* [8] proposed a plan-driven methodology that is massive in size and only suitable for standard software industry with long-term objectives. The AZ-Methodology of Azeem *et al.* [37] integrates software quality factors based on the standardization guidelines but ignores the scope of a distributed working place [38]. A monolithic ontological methodology suggests applying descriptive logic for information specification and quality control [15], but it ignores distributed project management factors. A software project that explicitly specifies its information with descriptive logic or control language reduces complexity, ambiguity, language barriers, and improves formal practice [39]. A sustainable quality methodology promotes considering sustainability factors (effective algorithms and data structures) during development, practicing sustainability in the software industry, and promoting a sustainable product [9]. The ontological framework supports a software project to run efficiently and reduces project complexity [18]. Information specification for big data or data science projects can reduce complexity through ontological methodology [40]. A distributed working environment needs standard documentation so that team members can easily understand from another corner of the world. A control language can reduce ambiguity for a distributed working place. Nowadays, the distributed development environment improves sustainability in software development, testing, and maintenance. This research proposes a framework to minimize the challenges of the distributed software development environment.

III. Proposed Framework

The proposed model is illustrated by Fig. 3 that separates the virtual managerial functionalities of a software development process from physical management system. Inhouse project management approach distributes the tasks to the virtual teams that are controlled by conventional documentation, guidelines, and policy procedure of the organization. The proposed framework is not designed for total virtual management environment to reduce the complexity. The manager's roles of a software firm are remaining as task of a physical workplace where top management can guide to the manager.

The distributed approach shows importance on explicit documentation for eliminating ambiguous information [15, 38, 39] so that it can reduce challenges of generic language. Descriptive logic or control logic is recommended in virtual projects to specify the tools for design, instruction generation, documentation and specification of software's information: user requirement specification, documentation of software design, and code convention. When a software designer uses control logic rather than generic language it becomes easy and clearer to the technical persons, analysts, designers, developers, testers, and manager [9]. So this research shows the way of software information specification by descriptive logic.

Descriptive logic are formal and applied to present domain knowledge and relevant reasoning with two forms of terminologies called Knowledge representation and model logic, respectively. There are different types of descriptive logic and they are varied by expressibility as well as reasoning complexity. In knowledge-based applications An individual is a constant, A concept is a unary relation and a role is a binary relation. It includes syntax, semantics, ABox, TBox, inference relationships, and formalism. Specification terminology is represented.

A. Information Speccification by Descriptive Logic

Set: A Project P consists of elements and can be represent as, {stakeholder, process, activities}⊆P. Set: S for stakeholders has external stakeholders: { {client, user, sponser, government}⊆External_stakeholder, and internal stakeholders can be representd as {system analyst, designer, developer, manager}⊆Internal_stakeholder }}⊆S. Moreover, we can present a client by the expression: client∈ External_stakeholder, and end_use∈user (end user is an element of user). Similarly we can represent other components like: individual_government_agency ∈ government as well as, system analyst ∈ Internal_stakeholder, individual_manager∈manager and so on. Set: V_S for virtual stakeholder where (designer, developer, software_tester) ⊆S. Set: Pe for people where {S}⊆Pe and individual stakeholder ∈S. Set: M for manager where {{{ Responsibility, leadership, skill}⊆M} ⊆S}⊆Pe; and each responsibility∈m, each leadership quality∈m, and unique skill ∈m. The same representation can be presented for each elemnt like stakeholder, task, and process.

B. Relationship of the Project's Inormation

Set T: If T represents any task of a project, { requirement analysis, design, development, test, deploy}⊆T. Set: R for requirement where {functional, non-functional}⊆R that related to others elements can be mention as follows: {individual_input∈input_set, individual_process∈process_set, individual_output∈ output _set ⊆ functional_requirement and {accuracy∈ quality_factor, efficiency∈quality_factor, portability ∈quality _factro,}⊆nonfunctional_requirement. Set: naming convention for data, object, class, and files. If I for input, O for output and Pr for Process, the ∀I.Pr and ∀O.Pr, Where (Pr, I):O and individual presentation is ∃Ii .Pri means input Ii has existed to Process Pri. For (Pri,Ii):Oi , Pri is the process the generates output OI which is related to Ii. The relationship of DataField (DFi) and DataTable (DTi) is IsBelongTo (DFi,DTi). Set: Flow of content and management procedure (Fig. 3) and develop a centralized application with modules: task distribution, accepting feedback, formal and logical document development, and managerial functionalities. Flow management shows the logical connection between physical and virtual management.

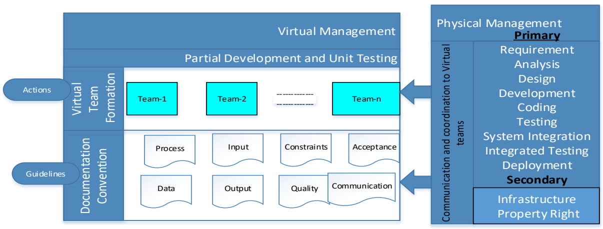


Figure 3. Distributed software project management framework.

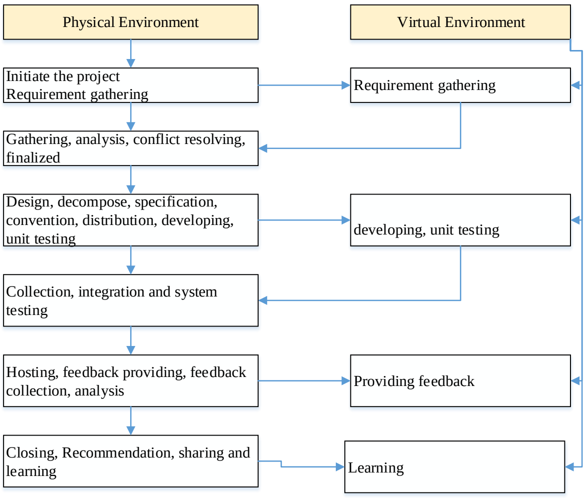

Physical Environment
Virtual Environment
Initiate the project
Requirement gathering
Requirement gathering
Gathering, analysis, conflict resolving, finalized
Design, decompose, specification, convention, distribution, developing, unit testing
developing, unit testing
Collection, integration and system testing
Hosting, feedback providing, feedback collection, analysis
Providing feedback
Closing, Recommendation, sharing and learning
Learning

Figure 4. Framework execution procedure.

Fig. 4 distributes the tasks into two groups: task of physical environment and virtual environment. All phases of a software project is initiated by physical environment and distributed to virtual environment with required documentation. Physical environment also collects the solutions of tasks from the virtual employees and performs integration. When one phase is completed it initiates to distribute the tasks for the next phase. This is well defined and secuential process of a project management.

IV. Framework Assessing Criteria

The main aim of the paper is to reduce ambiguity in a virtual software project management system through a formal management approach. Moreover, change management and feedback consideration besides, task allocation are important criteria too. This paper considered the latest three technological trends in IT projects to develop evaluation criteria. The criteria were developed after discussion with the experts of the developing teams. These projects require more detailed information specification and involve higher complexity in management [28]. If any project in these categories can be effectively managed in a virtual environment, we assume that other types of software development projects will also be easier for the firm.

1) IoT projects: IoT projects are embedded in two domains: a) specific practices, where one option is selected by comparing different IoT options, and b) asset-integration architecture, which is used to identify the devices, gateways, and services and their responsibilities. Providing feedback, managing change, and recording documentation can be challenging when working on multiple project streams simultaneously.

2) Blockchain project: Blockchain projects are highly influenced by security and uncertainty, as well as by issues such as lack of scalability, difficulty with integration, and legacy systems. Proper preparation, risk assessments, and documentation are important in project execution.

3) AI project: Main challenges with data quality, case-specific learning, integration, and legal. AI project requires a strategic approach, setting objectives, identifying performance indicators, and tracking machine learning performance.

Project estimation and methodology selection are important tasks to complete before implementation. Project estimation and methodology selection are important tasks to complete before implementation, especially when running a project online. Whether a software project succeeds or not also depends on the estimate and the plan. A virtual project certainly reduces the cost and time to achieve the commercial objective. There are many influential factors to consider in project estimation. For example, Govil and Sharma [41] mention 36 common factors that support the quality of the software product and effective project management practices. Anjana and Ganga [42] suggested a new measure for the object-oriented software project which could be used to implement distributed project estimation. We choose the methodologies to compare with the proposed model which is well known and widely used in previous studies [8, 15].

Agile methodologies are commonly used for blockchain and IoT projects, while scrum is the most popular methodology in the agile family. Waterfall and agile methodologies are suggested for managing AI projects. In Europe, PRINCE2 is the most popular and all government project is maintained by PRINCE2 in the UK. The proposed model is being compared to the aforementioned models with respect to the influential factors [8, 26] of technological advancement: online management, standard documentation, formal management, security, integration, scalability, and sustainability.

V. Framework Evaluation

From the agile family of methodologies, the scrum methodology was selected because of its popularity among users. It has a sprint review structure that allows multiple iterations for quality control. Its regular meetings improve communication and reduce misunderstandings, making a project more manageable. Scrum teams can work to improve the quality within a time-constrained environment, but ensuring proper security can be difficult. Moreover, there is no scope to enhance the scalability and backlog of the project with the scrum methodology. The scrum methodology lacks a clear vision for virtual management and documentation practices. The waterfall methodology is rigid with its phases and is not suitable for large, risk-oriented projects. n a distributed environment, implementing a fixed phase approach can be difficult. PRINCE2 is a generic methodology, and additional work is needed to adapt it to a software project. It can allow for distance management. So system management needs to re-structure PRINCE2 for their own project. Table I shows that existing methodologies have limited potential for improvement through remote working.

This research includes four evaluation criteria including scope of the virtual project, information specification of the project, virtual project management, and scalability issues of a virtual project. The evaluation criteria were selected based on the objective (virtual project management) of the study from previous works [8, 9, 15]. A project moves towards virtual management to improve business goals, and a manager must clarify the scope and information with control logic so that any technical person can understand and easily convert a program from design. Distributed developers can only use specified information so that a manager can easily integrate the distributed segments of an application into a program. This documentation will support further scalability actions.

TABLE I. A COMPREHENSIVE COMPARISON STUDY WITH THE MOST POPULAR PROJECT MANAGEMENT APPROACHES (SYSTEM DEVELOPMENT LIFE CYCLE AND PROJECT MANAGEMENT METHODOLOGIES)

Plan-driven
Scope: *"Plan-driven methodologies are based on waterfall structures that are rigid in architecture and have limited adaptability. Iterative or interactive methodologies enhance the review process but do not facilitate virtualization.* ***Documentation.*** *Completing the documentation with requirement analysis that is in an earlier stage does not specify the structure of the documentation.* ***Management:*** *Traditional management systems distribute the work after the planning phase and everyone is busy with their own responsibility but there is no formal procedure of communication without subsequent phase relationship.* ***Scalability:*** *After requirement analysis, it is recorded and managed by documentation. This could enhance the future work but with no guarantee of re-usability of resources to scale up. This creates the lack of structured record-keeping practice in project management.*
Agile Methodologies
Scope: *Agile methodologies aim to complete a task within minimum time and follow a constraint time plan. So it can accept the scope to access experts from the corner of the world. However, it needs a virtual communication tool.* ***Documentation:*** *Although Agile methodologies do not prioritize documentation, the virtual environment requires unambiguous documentation..* ***Management:*** *An Agile team is responsible for accomplishing tasks on time with a high spirit of teamwork, which demands a good team leader. Such leadership can help manage virtual teams as well.* ***Scalability:*** *The lack of formal documentation practice in Agile methodologies limits its ability to work on scalability.*
PRINCE2
Scope: *PRINCE2 inspires access tools and techniques for virtualization to manage any project but there are no specific guidelines for the software project.* ***Documentation:*** *It has generalized guidelines for documentation practice but software development needs controlled language that is more specific.* ***Management:*** *PRINCE2 has a structured management formation than can enjoy virtualization with sufficient tools and techniques.* ***Scalability:*** *PRINCE2 does not provide specific guidelines for enhancing software scalability, so teams using this methodology must adapt its functionalities to address scalability issues and improve the project's scalability scope.*
PROPOSED FRAMEWORK
Scope: *It is specialized for distributed software project management that considers required actions, process, procedure and suggests to utilize applications. So it has its advantages over other system development approaches for virtual project management.* ***Documentation:*** *It suggests to use descriptive logic rather than generic language that will avoid ambiguous documentation. Furthermore, it will improve the reusability of the document and the programmer can easily convert logic to code.* ***Management:*** *It has different management views for the physical and virtual working environment respectively. Formal communication and interaction process for both modes.* ***Scalability:*** *It has soft descriptive logic, documentation which easily updates, modify and re-usability helps to enhance an existing project.*

The proposed framework is presented to the experts besides providing an execution guideline and a poster for clarification. We gave them six months duration (April to September 2000) to practice in their projects. The data is collected through google survey file. Five different experts who have mixed experiences on freelancing, overseas projects, and the local development team submitted their evaluation. They have 2–7 years of experience of working in development and project management with multiple methodologies. The questionnaire is presented in Appendix-A. There are many multicriteria decision-making methods, we applied Grey Rational Analysis (GRA) to evaluate similar types of studies that are carried out by group analysis [38]. In this study, we applied the analytical hierarchy process (AHP) which is performed by comparing pairs which gives a more focused reflection of an evaluator [8, 15]. The final result is confirmed by the average rating of the experts.

Analytical Hierarchy Process (AHP) is used for the paired comparison of each factor because pair comparison is easire than group comparison of GRA. It reduces the complexity of the respondents. It's pair comparison scale consists of odd numbers from 1 to 9 with a ranking factor: Extreme Favors (EF)= 9, Very Strong Favors (VSF)= 7, Strongly Favors (SF)= 5, Slightly Favors (SlF)= 3, Equal (E)= 1. Among the five outcomes from five experts, the median is considered as the AHP value for the particular measuring factor. A Reciprocal matrix is developed based on the feedback and respective inverse value. For example, 1/3 is the reciprocal value of 3 (Aij=1/Aji). Fig. 5 shows the reciprocal matrix and priority rank that was developed by normalizing (cell value/total of each column) and summarizing row values.

Fig. 5 reflects the quantitative pair comparison in four different parameters based on the Table I and comparative study. The proposed model is more advanced in ranking for (a), (b), and (c); while PRINCE2 is much better in criteria (d). The primary goal of the proposed method is achieved by partial virtualization and information specification with descriptive logic (e).

Reciprocal Matrix					Priority	
	Pm	Wf	Sc	P2		
Pm	1	9	7	5	Pm	2.5627729
Wf	1/9	1	1/3	1/5	Wf	0.1998319
Sc	1/7	3	1	1/3	Sc	0.4042057
P2	1/5	5	3	1	P2	0.8331895

(a)Sope of virtualization

	Pm	Wf	Sc	P2		
Pm	1	7	7	5	Pm	2.5548951
Wf	1/7	1	1	1/3	Wf	0.313037
Sc	1/7	3	1	1	Sc	0.5468032
P2	1/5	3	1	1	P2	0.5852647

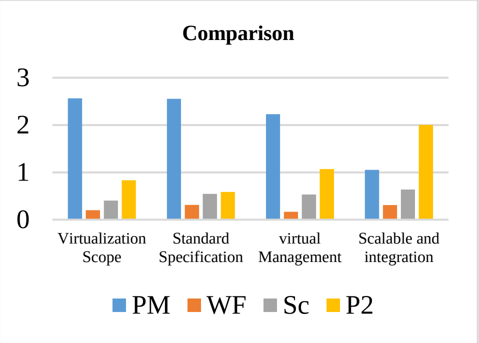
(b)Specification and documentation

	Pm	Wf	Sc	P2		
Pm	1	9	5	3	Pm	2.23089
Wf	1/9	1	1/5	1/7	Wf	0.1666761
Sc	1/5	5	1	1/3	Sc	0.5320581
P2	1/3	7	3	1	P2	1.0703757

(c) Scope of virtual management

	Pm	Wf	Sc	P2		
Pm	1	3	3	1/3	Pm	1.0519481
Wf	1/3	1	1/3	1/5	Wf	0.3073593
Sc	1/3	3	1	1/3	Sc	0.6363636
P2	3	5	3	1	P2	2.004329

(d) Scalable and Integration



(e) Comparison graph

Figure 5. Comparative analysis by AHP in four selective domains (a,b,c,d) and visualization (e).

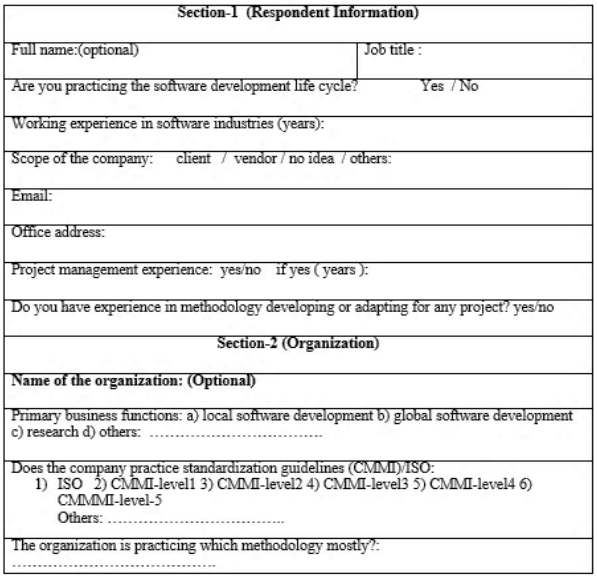
This model virtualizes for requirements gathering, accepting feedback, and module wise problem-solving and testing. It keeps central management to the physical body so that it can control in-house functionalities as well as remote activities. Centralized management is recommended for reducing task distribution conflicts, minimizing documentation conflicts, reducing ambiguity, and resolving conflicts. Guidelines, procedures, policies, and naming conventions are maintained for input, process, and output to ensure uniformity in design and development. Documentation based on descriptive logic is implemented to minimize language skill constraints. Moreover, technical terminologies and their notation are comparatively easy for a technical person to understand for project management, system specification, and software functionality presentation. Based on expert feedback, the proposed framework includes the main criteria required to minimize the challenges of virtual software project management.

VI. Conclusion and Future Work

It is a modified methodology with additional features of any of the existing methodologies. It suggests a way to improve software project management by accepting the technological advancement of virtualization. In addition, upcoming challenges for technological advancement in the nature of the software projects could be addressed by explicit specification, formal documentation, and an effective management strategy. It shows that descriptive logical expression is one way to standardize documents and specifications, reducing generic language skill barriers. Furthermore, the utilization of global opportunities is formalized in the software industry.

This is a simplified model that separates virtual management functionalities from traditional approaches. It omits the explanation of traditional phases to allow adaptation to any existing traditional approach. The standard software firms physically exist with business goals and the proposed model avoids the complexity of a completely virtual process. Online procurement management can manage ad-hoc or specialized software, but it is not logical for a standard organization to outsource everything without utilizing organizational resources. The management software should be much more complex, with additional functionalities, integration opportunities, monitoring, and control strategies. E.g., this model only performs unit testing in the virtual environment and an integrated system is performed in the organization, but it will be much more complex when different modules collect and integrates and perform system testing. This paper has a scope to develop a methodology for completely virtual project management for soft products.

Appendix A: Frame Work Evaluation



Section-1 (Respondent Information)

Full name:(optional)

Job title :

Are you practicing the software development life cycle? Yes / No

Working experience in software industries (years):

Scope of the company: client / vendor / no idea / others:

Email:

Office address:

Project management experience: yes/no if yes (years):

Do you have experience in methodology developing or adapting for any project? yes/no

Section-2 (Organization)

Name of the organization: (Optional)

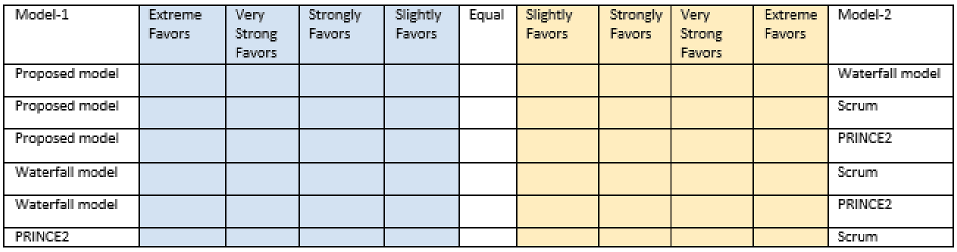
Primary business functions: a) local software development b) global software development c) research d) others: ………………………………

Does the company practice standardization guidelines (CMMI)/ISO:
1) ISO 2) CMMI-level1 3) CMMI-level2 4) CMMI-level3 5) CMMI-level4 6) CMMMI-level-5
Others: ………………………………..

The organization is practicing which methodology mostly?:
…………………………………………

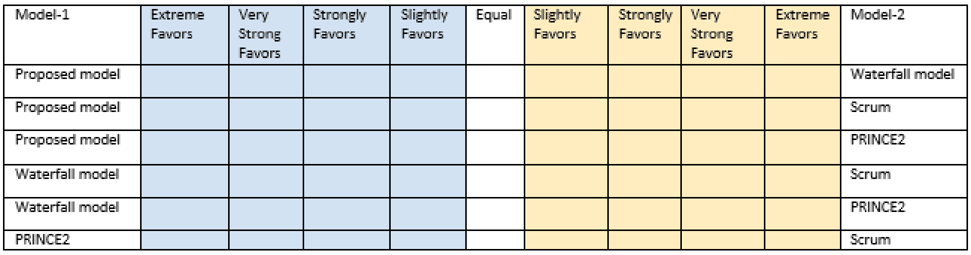
Section III (Put tick √ for you best choice) for example: If you found that the proposed model is equal to waterfall model for a criteria you will tick respective/ white cell (middle column and second row). You you feel that laft side model is more prefarble than right side model you shold tick in respective blue cell; or if you feel that write side model is more preferable you will tick to the respective yellow cell.

Q1: Perform pair comparison for "the virtual project managmet scope" that is maintained by the modles.



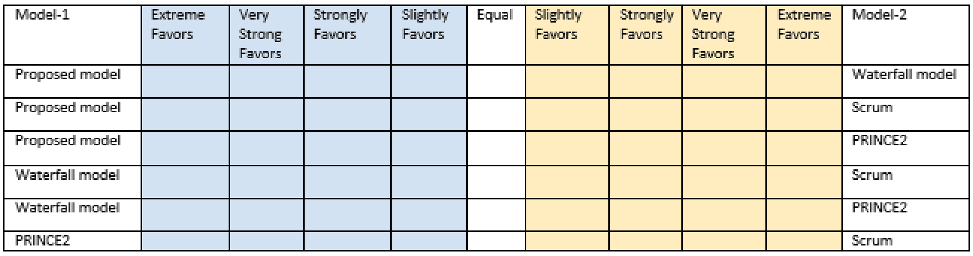
Model-1	Extreme Favors	Very Strong Favors	Strongly Favors	Slightly Favors	Equal	Slightly Favors	Strongly Favors	Very Strong Favors	Extreme Favors	Model-2
Proposed model										Waterfall model
Proposed model										Scrum
Proposed model										PRINCE2
Waterfall model										Scrum
Waterfall model										PRINCE2
PRINCE2										Scrum

Q2: Perform pair comparison for the “ documentation practice in virtual project” that is maintained by the models.



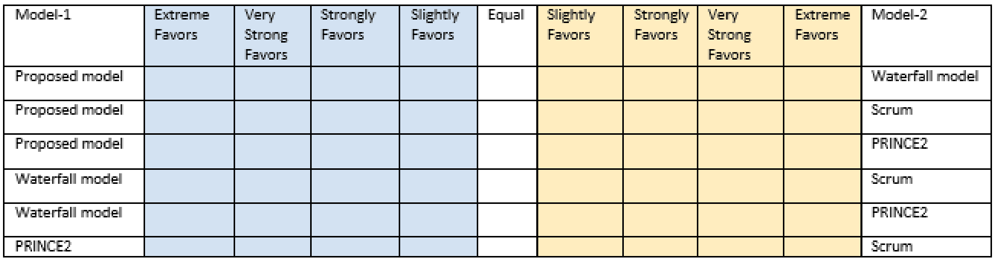
Model-1	Extreme Favors	Very Strong Favors	Strongly Favors	Slightly Favors	Equal	Slightly Favors	Strongly Favors	Very Strong Favors	Extreme Favors	Model-2
Proposed model										Waterfall model
Proposed model										Scrum
Proposed model										PRINCE2
Waterfall model										Scrum
Waterfall model										PRINCE2
PRINCE2										Scrum

Q3: Perform pair comparison for “the virtual management facility” that can practice by the models.



Model-1	Extreme Favors	Very Strong Favors	Strongly Favors	Slightly Favors	Equal	Slightly Favors	Strongly Favors	Very Strong Favors	Extreme Favors	Model-2
Proposed model										Waterfall model
Proposed model										Scrum
Proposed model										PRINCE2
Waterfall model										Scrum
Waterfall model										PRINCE2
PRINCE2										Scrum

Q4: Perform pair comparison for “the scalability and enhancement facility” that can integrated to the models.



Model-1	Extreme Favors	Very Strong Favors	Strongly Favors	Slightly Favors	Equal	Slightly Favors	Strongly Favors	Very Strong Favors	Extreme Favors	Model-2
Proposed model										Waterfall model
Proposed model										Scrum
Proposed model										PRINCE2
Waterfall model										Scrum
Waterfall model										PRINCE2
PRINCE2										Scrum

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

For the titled manuscript there are four authors where each author has contributed in their own way to support and obtain better results for the given work. Kamal is a primary author who has taken most of the responsibility of performing and conceiving the design and revising the manuscript. Raza holds the responsibility of collecting the data and writing the reflections for the literature reviewed. Salman wrote the paper and worked on grammar. Aziz is the supervisor. All the authors had approved the final version.

Acknowledgment

The authors wish to thank the participants who took part in the study for their valuable inputs and suggestions.

References

[1] B. Flyvbjerg and A. Budzier. (2011). Why your IT project may be riskier than you think. *Harvard Business Review*. [Online]. Available: https://hbr.org/2011/09/why-your-it-project-may-be-riskier-than-you-think#comment-section

[2] Project Management Institute. (2017). Success rate rise, performing the high cost of low performance. [Online]. Available: https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2017.pdf

[3] The Standish Group International Inc. (2018). The Standish Group Report: CHAOS. [Online]. Available: https://www.standishgroup.com/sample_research_files/DemoPRBR.pdf

[4] Geneca. (2017). Why up to 75% of the project will fail? [Online]. Available: https://www.geneca.com/why-up-to-75-of-software-projects-will-fail/

[5] T. O. A. Lehtinen, M. V. Mäntylä, J. Vanhanen, J. Itkonen, and C. Lassenius, “Perceived causes of software project failures — An analysis of their relationships,” *Information and Software Technology*, vol. 56, no. 6, pp. 623–643, 2014.

[6] R. N. Charette, “Why software fails,” *IEEE Spectrum*, vol. 42, pp. 36–43, 2005.

[7] Project Management Institute. (2014). The Project Management Office: Aligning Strategy & Implementation. [Online]. Available: https://www.pmi.org/-/media/pmi/documents/public/pdf/white-papers/pmo-strategy-implement.pdf?v=21d05146-7d8c-46c9-aff8-5b472d1c7509

[8] K. U. Sarker, A. B. Deraman, R. Hasan, and A. Abbas, “A 4-Layered Plan-driven Model (4LPdM) to improve software development,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 8, pp. 367–376, 2022, doi: 10.14569/IJACSA.2022.0130868

[9] K. U. Sarker, A. B. Deraman, R. Hasan, A. Abbas, “SQ-Framework for improving sustainability and quality into software product and process,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 9, pp. 69–78, 2020, doi: 10.14569/IJACSA.2020.0110909

[10] S. Nauman and S. Iqbal, "Challenges of virtual project management in developing countries," in *Proc. the 2005 IEEE International Engineering Management Conference*, 2005, pp. 579–583.

[11] S. Gera, "Virtual teams versus face to face teams: A review of literature," *IOSR Journal of Business and Management*, vol. 11, no. 2, pp. 1–4, 2013, doi: 10.9790/487X-1120104

[12] A. Ledwith and P. Ludden, "A typology framework for virtual teams," Project Management Institute, 2016.

[13] A. Abyad, "Globalization challenges in project management," *Middle East Journal of Business*, vol. 12, no. 4, pp. 10–19, 2017, doi: 1EJB.2017.93085

[14] CodingSans. (2019). State of Software Development. [Online]. Available: https://codingsans.com/state-of-software-development-2019

[15] K. U. Sarker, A. B. Deraman, R. Hasan, A. Abbas, B. Shah, and A. Ullah, "Monolithic Ontological Methodology (MOM): An effective software project management approach," *Journal of Engineering Research (JER)*, vol. 10, no. 2A, pp. 30–40, 2022.

[16] M. Khari, P. Kumar, and G. Shrivastava, "Test optimisation: An approach based on modified algorithm for software network," *International Journal of Advanced Intelligence Paradigms*, vol. 17, no. 3–4, pp. 208–237, 2020, doi: 10.1504/IJAIP.2020.110182

[17] M. Khari, A. Sinha, E. Herrerra-Viedma, and R. González-Crespo, "On the use of meta-heuristic algorithms for automated test suite generation in software testing," in *Toward Humanoid Robots: The Role of Fuzzy Sets*, Springer, 2021, pp. 149–197.

[18] M. Khari, D. B. Mishra, B. Acharya, and R. G. Crespo, "Optimization of automated software testing using meta-heuristic techniques," in *Proc. the International Conference on Computer Networks, Big Data and IoT*, 2022, pp. 9–11.

[19] A. Jain, D. K. Tayal, M. Khari, and S. Vij, "A novel method for test path prioritization using centrality measures," *International Journal of Open Source Software and Processes (IJOSSP)*, vol. 7, no. 4, pp. 19–38, 2016, doi: 10.4018/IJOSSP.2016100102

[20] A. Takyar. (2019). Software development trends to watch for in 2019. [Online]. Available: https://themanifest.com/development/software-development-trends-watch-2019

[21] S. Porru, A. Pinna, M. Marchesi, and R. Tonelli, "Blockchain-oriented software engineering: challenges and new directions," in *Proc. 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*, 2017, doi: 10.1109/ICSE-C.2017.142

[22] R. Motta, K. Oliveira, and G. Travassos, "On challenges in engineering IoT software systems," in *Proc. the XXXII Brazilian Symposium*, 2018, pp. 42–51, doi: 10.1145/3266237.3266263

[23] H. Belani, M. Vuković, and Ž. Car, "Requirements engineering challenges in building ai-based complex systems," arXiv preprint arXiv:1908.11791, 2019.

[24] H. Madurika and G. Hemakumara, "Gis based analysis for suitability location finding in the residential development areas of greater matara region," *International Journal of Scientific & Technology Research*, vol. 4, no. 8, pp. 96–105, 2015.

[25] S. Zakeri, "Ranking based on optimal points multi-criteria decision-making method," *Grey Systems: Theory and Application*, 2018, doi: 10.1108/GS-09-2018-0040

[26] W. Sałabun, "The characteristic objects method: A new distance-based approach to multicriteria decision—Making problems," *Journal of Multi-Criteria Decision Analysis*, vol. 22, no. 1–2, pp. 37–50, 2015.

[27] J. Rezaei, "Best-worst multi-criteria decision-making method: Some properties and a linear model," *Omega*, vol. 64, pp. 126–130, 2016, doi: 10.1016/j.omega.2015.12.001

[28] A. Mardani, A. Jusoh, and E. K. Zavadskas, "Fuzzy multiple criteria decision-making techniques and applications—Two decades review from 1994 to 2014," *Expert Systems with Applications*, vol. 42, no. 8, pp. 4126–4148, 2015, doi: 10.1016/j.eswa.2015.01.003

[29] A. Mardani, E. K. Zavadskas, K. Govindan, A. A. Senin, and A. Jusoh, "VIKOR technique: A systematic review of the state of the art literature on methodologies and applications," *Sustainability*, vol. 8, no. 1, p. 37, 2016, doi: 10.3390/su8010037

[30] G. Haseli, R. Sheikh, and S. S. Sana, "Base-criterion on multi-criteria decision-making method and its applications," *International Journal of Management Science and Engineering Management*, vol. 14, no. 2, pp. 94–103, 2019, doi: 10.1080/17509653.2019.1633964

[31] B. Blanchard and W. Fabrycky, *Systems Engineering and Analysis*, 4th ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2011.

[32] A. A. Adenowo and B. A. Adenowo, *Software Engineering Methodologies: A Review of the Waterfall Model and Object-Oriented Approach*, 2013.

[33] C. Larman and V. R. Basili, "Iterative and incremental developments. A brief history," *Computer*, vol. 36, no. 6, pp. 47–56, 2003, doi: 10.1109/MC.2003.1204375

[34] D. Gurendo. (2015, October). Software Development Life Cycle (SDLC): Spiral model. XB software. [Online]. Available: https://xbsoftware.com/blog/software-development-lifecycle-spiral-model/

[35] S. Mathur and S. Malik, "Advancements in the V-model," *Int. J. Comput. Appl.*, vol. 1, no. 12, pp. 29–34, 2010, doi: 10.5120/266-425.

[36] P. Kruchten, *The Rational Unified Process—An Introduction*, 2000, Reading, MA, USA: Addison-Wesley.

[37] M. A. Akbar, J. Sang, A. A. Khan, N. Fazal-E-Amin, M. Shafiq, S. Hussain, H. Hu, M. Elahi, and H. Xiang, "Improving the quality of software development process by introducing a new methodology—AZ-model. *IEEE Access*, vol. 6, pp. 4811–4823, 2018, doi: 10.1109/ACCESS.2017.2787981

[38] K. U. Sarker, A. Deraman, R. Hasan, and A. Abbas, "Explicit specification framework to manage software project effectively," *Indian Journal of Science and Technology*, vol. 13, no. 36, 3785–3800, 2020, doi: 10.17485/IJST/v13i36.1244

[39] K. U. Sarker, A. B. Deraman,and R. Hasan,"Ontological practice for software quality control," *International Journal of Business Information Systems*, vol. 34, no. 3, pp. 355–372, 2020, doi: 10.1504/IJBIS.2020.108650

[40] K. U. Sarker, A. B. Deraman, R. Hasan, and A. Abbas, "Ontological practice for big data management," *International Journal of Computing and Digital Systems*, vol. 8, no. 3, pp. 265–272, 2019, doi: 10.12785/ijcds/080306

[41] N. Govil and A. Sharma, "Estimation of cost and development effort in Scrum-based software projects considering dimensional success factors," *Advances in Engineering Software*, vol. 172, 103209, 2022.

[42] A. Gosain and G. Sharma, "A new metric for class cohesion for object oriented software," *The International Arab Journal of Information Technology*, vol. 17, no. 3, 2020.