# A 4-Layered Plan-Driven Model (4LPdM) to Improve Software Development

Kamal Uddin Sarker[1]
Department of Computer Science
American International University Bangladesh
Dhaka, Bangladesh

Aziz Bin Deraman[2]
Faculty of Ocean Engineering Technology and Informatics
University Malaysia Terengganu
Terengganu, Malaysia

Raza Hasan[3]
Department of Computing and IT
Global College of Engineering and Technology
Muscat, Oman

Ali Abbas[4]
Department of Computing
Middle East College
Muscat, Oman

***Abstract*—Quality is the degree of excellence of a product and one of the most important factors of software projects that mainly defines user satisfaction and success of the project. Software methodologies represent a variety of tasks, processes, and roles to manage time, cost, and quality. The invention, innovation, and diffusion for technological advancement creates challenges of software projects, thus several existing methodologies albeit with limited scope. A software product is highly influenced by the latest technology and distributed project management opportunities. Management issues are introduced for a virtual project management environment when resource persons are in another corner of the world. To resolve the problem, this research presents a new software project management model (4-LPdM) with alternative actions and practices to effectively manage. The model was presented to 20 different organizations and 29 respondents gave feedback who had experience between 1-16 years in multiple sections of software engineering. The model is evaluated based on the factors of advanced PMBOK 4.0 (scope, cost, quality, resource, risk, plan) and two (management, sustainability) additional features according to the demand of experts. This research illustrates statistical analyses to examine the significance of the proposed model besides a comprehensive comparative study of the traditional methodology.**

***Keywords—Software development methodology; project management; 4-layered plan-driven model; quality factors; sustainability***

## I. INTRODUCTION

Software engineering is continuously upgrading with its area of application and research with the advancement of technology by accepting new opportunities and overcoming challenges. The capability and features of the software are increasing with efficient data handling and adding smart functionalities. Smart devices, industrial automation, artificial intelligence equipment, and digital business concepts enhance opportunities for software industries and researchers. An industry expert or researcher can work from any corner of the world through a distributed working environment. Software engineering process management methodologies are improving since the commence of the waterfall model (1956) and followed by various normalized water-fall models (1956-1985), spiral (1986), scrum (1990-1995), rational unified process (RUP, 1996-1998), extreme programming (XP, 1999), agile manifesto (2001), lean (2003), 5th value agile manifesto (2008), DevOps (2009), and Kanban (2010) [1]. Besides methodologies, a set of software quality models is working to improve the quality of the software are McCall's Quality Model (1977), Boehm's Quality Model (1978), IEEE Quality Model (IEEE Std. 729-1983), ISO 9126 Quality Model (1991-2011, ISO/IEC 9126) [2-3]. Plenty of standardization organizations are also working to improve the software project management approach by providing guidelines [4]. But till now 83.9% of information technology (IT) projects completely failed (stop without delivery) or partially failed (compromise with quality) according to Standish Group CHAOS report 2018 [5]. Harvard business review reports that one-sixth of projects run over budget by 200% and 5-15% of projects are failed [6]. Moreover, the project management institute noticed 80% were completed on time without significant wastage of money but the quality is very poor [7] and 75% of IT executives found that their projects were doomed usually from the beginning [8].

So, practitioners, researchers, standardization bodies are working to improve the quality of software products and deduct project failure rates. Digital transformation introduces newer challenges in software industries. Rapidly upgrading technology, increasing functionalities, and changing infrastructure make the software project crucial [9]. Artificial intelligence, cross-platform, Internet of Things, blockchain, continuous development and deployment, progressive web applications, and low-code developments are new trends in software industries [10] that demand more specification of software project information. Near future software engineering becomes close to system engineering that demands structured methodology and architecture for reliable system development with variability features [11]. Furthermore, a distributed software project management system introduces new challenges with diversity culture, different time zone, language barriers, lack of collaboration and communication, trust and ownership of intellectual property, unjustified requirement specification, integration hassle, and lack visionary practice

[12]. A methodology accelerates the project execution process, managing resources, enhancing formal practice, improving sustainability, and ensuring projects quality. Nowadays, systems are improving from on-premises to cloud infrastructure, microcontroller based to industry IoT automation with machine learning, deep learning or reinforcement learning and faces big data challenges (volume, variety, veracity, velocity). Project migration inherits a high grade of complexity with the broader challenge of data collection, specification, sharing, transformation, and analysis when a new technology is being adopted. A well-structured methodology keeps maintainability, portability, and scalability scope with standard guidance and documentation practices.

Current issues in software projects and roles of methodology to address the challenges and make a project successful are mentioned in the introduction. The article is followed by a literature review that consists of reasons for software project failure and the importance of upgrading methodologies. Section III proposes a methodology that is illustrated by Fig. 1 and elaborates on its functionalities in the sub sections. The research methodology discusses ways of evaluation of the proposed methodology. The proposed methodology brings quality of process and product in an architecture. It consists of process, task, and people under the quality control framework. Section V discusses detailed outcome of the study and concludes in section VI with limitation, future work and remarks of contribution.

## II. Literature Review

Project management approaches are improving by introducing new inventions in process, tools, and management. Software process management methodologies involve the invention to complete projects effectively. Software development endeavors are affiliated with the practice of a creative project management approach. Software projects are regularly related to innovation in management to overcome the challenges of the fast and dynamic changing of technologies and focus on the best services and products. A higher degree of creativity and flexibility are required in practice with the innovative process of methodology. This section consists of short literature on the reasons for project failure, the role and progress of methodology, and the existing gap.

### *A. Software Project Failure and Methodology*

A project executes by a group of team members with distinguished responsibility that might be varied from organization to organization and project to project based on the type and nature of the project, mission and vision of the organization, and business goals. Potential stakeholders work in a team to make a project successful, but a significant number of projects fail to include its' users. A project commences with a determination to complete on time and budget, but it faces difficulties in the execution period with factors related to process, task or person. As a result, it becomes a challenging project and if the project team is unable to overcome issues it will fail. A project has a chance to face challenges by the wrong strategy of a project or organization [13], wrong or unrealistic planning [14], lack of stakeholders' support [15], and weakness of project management professionalism [16]. Discenza and Forman show the importance of adequate technical and non-technical resources, maintaining scarce resources, promoting effective communication, utilization of technical tools, and managing stakeholders' decisions by using operational metrics to make a project successful [17]. Kulish noticed that complexity of the design and code linearly related to the number of errors in a product. Time constraints of the project, human intervention factors, and miscommunication enhance projects' complexity [18]. Reasons for project failure are associated with people, technology, process, company, leadership, and business goals [19]. Uncertainty or risk is one of the most important reasons which makes a project fail [20] and it appears from stakeholders, technology, or nature. A model and methodology can set roles to develop an effective personality of stakeholders that could contribute to reducing the risk of a project [21]. A methodology approaches a systematic workflow and control of the project. It views on justified requirements, helps to estimate logical cost and effective hours, guides to incorporate change management, keeps standard documentation, ensures tracking on functional review, monitors, and controls on the project, allows backtracking if need (few cases), encourages formal communication among stakeholders, and helps to measure the size of the team of potential stakeholders [22].

### *B. Commonly used Methodologies*

Software process management methodologies have distinguished features and each one has special contributions in software engineering. The waterfall model is the first formal and most influential in software engineering [23]. It has sequential logical phases where one phase accepts feedback for the previous stage. The fundamental waterfall model is modified by overlapping functionalities of phases to utilize time and resource effectively [24]. But it is rigid with fixing requirements and confirming documentation at the earlier state of the project; moreover, users can share their suggestions only at the beginning. So, it is not appropriate for the projects where the requirements can change after execution of a project. The incremental approach is applied in software project management to bring more flexibility, where the client gets a solution part by part and the user can give feedback until the end of the project; iterative scope allows to give a partial solution and it should be updated by the several numbers of iterative feedback from users [25]. But iterative and incremental approaches have no standard architecture, so it is difficult to update and maintain the software. Too much user interaction in an iterative approach increases the scope of arguments. For example, the Spiral model is an iterative approach that is appropriate for high-risk and complex projects but difficult to implement time and cost constraint projects [26]. V-model integrates testing in all phases of the model to ensure the quality of the product [27], but not suitable for high risk, complex, object-oriented, and the project with moderate requirements. Parallel processing is initiated by the Rational Unified Processing model (RUP) which is time constraints iterative system, but it only focuses on functional requirements [28]. These are called heavyweight documentation-oriented plan-driven methodologies.

The agile approach brings innovation in software project management that helps to complete a task on time and does not support heavyweight documentation practice [29]. Agile methodologies are adopting a heterogeneous number of dynamic software projects where an organization's environment changes rapidly [30]. Agile is suitable for an organization that has a high probability to change management policy-procedure, tools and techniques, and working environment [30]. The agile manifesto is the foundation of agile families and scrum is popular in the agile family and it helps to manage complex projects by integrating creativity [31-32]. Extreme Programming (XP) allows customer interaction that operates by short iteration, Cristal methodology tailors' business goals and Agile Software Process (ASP) supports faster development [33]. Kanban method emphasizes business agility and realistic planning to deliver software products just-in-time [34]. Dynamic system development methodology (DSDM) emphasizes quality products in time constraints with limited iteration [35].

Build and fix is a methodology with lack of architecture and formal feedback which is reactive, and problems are fixed only when they occur. Waterfall is a liner approach where each phase is completed before continuing another one and there is lack in formal change management as well as feedback collection before completion of a project. V-shape is like waterfall but more concentrated on verification and validation in each phase and ignored risk analysis. Prototyping consists of three variations named rapid prototyping used for testing, evolutionary prototyping used for incremental improvements of the design, and operational prototyping improves the speed of production. Incremental consists of multiple cycle of development where entire process can restart any time that allows to change requirements and update a system. Spiral introduces risk analysis in iteration where new requirements are funneled and allows testing earlier. Agile is an umbrella of multiple methodologies that focuses on efficient and iterative development in an agile team.

Software methodologies could be divided into two major categories: plan-driven, and agile. Plan-driven are heavily weighted with documentation and rigid with a systematic approach. On the other side, agile methodologies are light weighted and have time constraints. Both approaches have pros and cons, such as plan-driven is process-oriented and it does not support requirement change frequently while agile methodologies face problems in maintenance and upgrading of a product.

Project management activities consist of methods, tools (e.g. Gantt Chart, network diagram, work breakdown structure), software (Microsoft Project), decision-making methods like feasibility study, risk analysis, and communication plan with collaborative tools: video conferencing [36]. The recent study (2019) of Walker and Lloyd noticed that project management work would be positive for non-routine workers by accepting advanced technology in the 2030s because of the border-free distributed working space [37].

### *C. Current Project Management Issues*

Unclear scope, time constraint, requirement changes, poor communications, managerial weakness, lack of formal practices, unrealistic resource allocation and planning, and insufficient testing are the common issues in software project management. But due to the technological advancement new challenges are appearing for AI, IoT, and big data projects. Technological projects need information specification for accuracy, reusability, scalability, and maintainability. Artificial intelligence applications, IoT software, and big data platforms use huge information that need to be specified explicitly by concept, role, and axiom [38-39] and descriptive logical or ontological presentation that improves ambiguity-free information for a shared domain [40]. Sarker et al. proposed a structure to develop and practice own methodology to consider effective internal and external stakeholders' participation for each project based on the user requirements and business goals [41]. Explicit information specification with ontology, descriptive logic, graphical presentation can reduce complexity of a project and improve communication among stakeholders [46]. A monolithically presented methodology with controlled language use to generalize the process of the methodology and improve integrated performance [44]. A methodology should consider sustainability factors into the product and encourages sustainability practices in project implementation [45].

Project management tools support to manage a project virtually and resource person can be distributed to the world that reduce office management cost, access talent from any corner of the world, and increase productivity; but need to overcome the challenges like virtual monitoring, multicultural team, trust on distrusted employee, and virtual communication [50].

4-LPdM proposed a formal approach, well defined framework, focus on information specification and four layered quality assurance that will reduce the issues of AI, big data, IoT, and distributed projects besides resolving regular issues.

## III. Proposed Model

4-Layered Plan-driven Model (4-LPdM) (Fig. 1) distributes the tasks into phases that are arranged in a logical order of waterfall architecture, but four transitions are specified called the layers. The first layer consists of requirement analysis and scheduling of the project, the second layer for in-depth design purposes, the third layer consists of coding and testing for both unit and system, while the fourth layer performs formal closing of the project. Furthermore, it shows the importance to specify the stakeholder, task, tools-techniques for each layer that helps to guide the model.

### *A. Phase-1: Requirement Specification*

The first layer (Fig. 1) consists of requirement collection and analysis from the respective sources where the users, experts, manager, and system analyst are the main key persons to accomplish the tasks. A user can share the visible requirements of functions that are required for the system. An expert will justify the requirements based on the demand of the market and competitors' values. A system analyst can support a manager to make decision for requirement fixing, technology selection, and cost benefit analysis. The 4-LPdM shows interest in recognizing, defining, measuring, and analyzing the requirements to sustain, improve, monitor and control. This phase maps business functions to the software process. Moreover, the planning phase has plenty of tasks that guide monitoring and control of the project. It suggests the utilization of tools, applications, and techniques to monitor, track, and control a project. This phase asks to develop stakeholder management policy, feedback accepting procedure, and communication plan. A good plan should consist of concrete goals, milestones, and tasks that are specified by date and tracking number. It also includes cost, time, and resource allocation for each task. Risk management activities are included to identify and take mitigation plans at the beginning of the project. Moreover, the feasibility study will help to measure the outcome of the project concerning customer/user requirements (functional and non-functional) and business goals of the vendor and client by cost-benefit analysis. The technical, operational, and ethical feasibility study will improve the acceptance of the system. This layer sets actions to fulfill the vision and mission of the project and achieve business goals. The quality team of this layer will review quality factors so that manager could incorporate required quality functions and information specification to lead IoT, big data, AI and distributed projects.

### *B. Phase-2: System Design*

Designers design the interface by incorporating accessibility and usability requirements; data design consists of standard specification, convention and controlled language for the project and management information; efficient databases consists of normalized tables with required integrity and constraints; control language (algorithm, pseudocode, descriptive logic, predicate logic for reducing misunderstanding), diagrams (context diagram, data flow diagram, entity-relationship diagram, sequence diagram, flowchart, etc.), interface, etc. In this phase (Fig. 1), feedback is expected from users, experts, and system analysts to ensure completeness and quality of the system. It keeps scope of the interaction of potential stakeholders. The design phase becomes more flexible than the traditional waterfall model because until the finalization of the design the user can change requirements. Customer interaction and satisfaction are extremely important for approving the design and passing the phase. System analysts can clarify requirements if required to the designers and they will finalize the architecture of the software. This phase concluded with structured documentation of earlier stages that could be shared through a distributed system, controlled by the manager, and flexible for reusability. This phase is more important to ambiguous free information specification for big data, IoT, AI projects and how to ensure effective communication in virtual project management.

### *C. Phase-3: Development and Deployment*

The third layer (Fig. 1) corresponds to the development, testing, and deployment phase which starts with coding. Effectiveness of the development and testing of a project is dependent on the quality of the design and explicit specification of data is mandatory for virtual or distributed project. The approved design of the previous phase is transformed into a programming notation according to a computer language. Programmers can easily convert the controlled language expressions of documentation and design materials. A standard notation can be easily converted to a program and support for test case generation. Unit testing is simultaneously performed by the quality control unit and developers. Experts are suggested to do template-based testing for accuracy and efficiency. A module consists of related units that are integrated and again tested by experts. The system is tested by quality control before handover to the customer, as well as after deployment, a short time of user training and testing is suggested so that users can use and manage the software effectively. This methodology focuses more activities in previous three phases to reduce complexity of fourth phase. Our proposed methodology supports to keep stand documentation that will improve maintainability, scalability, and portability for a project.

### *D. Phase-4: Formal Closing*

If the deployment test is satisfactory, then only it can be released for operation and the project enters in closing state (Fig. 1). A project teaches a lot of lessons to the stakeholders. Especially, mainstream project execution members get good experience related to technology, management, and communication. The closing layer of the project guides to analyze recently passed experiences so that team members can enhance their potential for future projects. In addition, it improves archiving quality to ensure reusability for further requirements; it helps to update project level or organization level policy procedure, and it is a good motivation for team members. 4-LPdM keeps formal documentation for reusability, maintainability, and scalability for an existing project, but it could be reuse for similar types of more future project and improve sustainability practice too. Project assessment and strategic planning will be easier for formal documentation practice.

### *E. Capability Study of 4-LPdM*

The proposed model is developed based on the demand of the near-future software project highly influenced by artificial intelligence, big data processing, embedded system, and significant risk for technological change. These systems require highly specified information by role, axiom, and concept. Moreover, there is a scope to hire (virtual environment) global talents for the project software projects. This section consists of the following capabilities of 4-LPdM.

### *F. Project Management Capability*

The 4-LPdM specifies the smallest unit of tasks for accurately measuring the size, functional dependencies, and complexity of the project. The project manager and his team define the errand and the execution team performs accordingly. The model prescribes to maintain standardization guidelines and values. Concerning requirements gathering, it gives importance to the client, system analyst, and experts from business and market domains. The design is developed based on the analysis of requirements that stop the scope creep possibility. Time, cost, and assets allocation ensure effective utilization at the micro-level. Critical risk mitigation and communication plans improve the awareness and responsiveness of the stakeholders. 4-LPdM blends the system development model and project management to improve the quality of products and processes. It gives the importance of utilizing tools and techniques for effort estimation, realistic plan development, stakeholder management, and formal documentation practice that will ensure an effective management process.

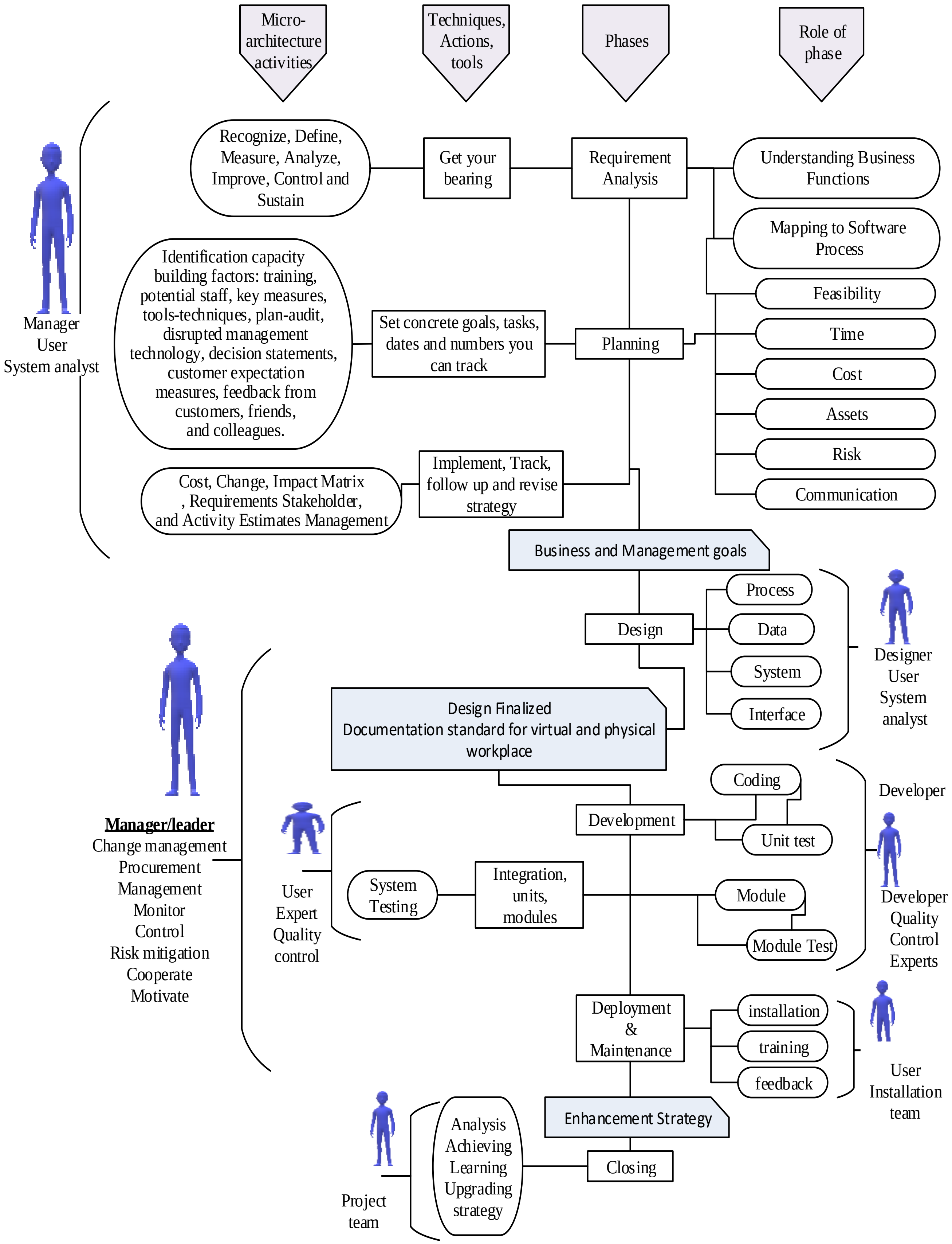
Micro-architecture activities
Techniques, Actions, tools
Phases
Role of phase
Recognize, Define, Measure, Analyze, Improve, Control and Sustain
Get your bearing
Requirement Analysis
Understanding Business Functions
Mapping to Software Process
Feasibility
Time
Cost
Assets
Risk
Communication
Manager
User
System analyst
Identification capacity building factors: training, potential staff, key measures, tools-techniques, plan-audit, disrupted management technology, decision statements, customer expectation measures, feedback from customers, friends, and colleagues.
Set concrete goals, tasks, dates and numbers you can track
Planning
Cost, Change, Impact Matrix , Requirements Stakeholder, and Activity Estimates Management
Implement, Track, follow up and revise strategy
Business and Management goals
Design
Process
Data
System
Interface
Designer
User
System analyst
Design Finalized
Documentation standard for virtual and physical workplace
Manager/leader
Change management
Procurement
Management
Monitor
Control
Risk mitigation
Cooperate
Motivate
Coding
Development
Unit test
Developer
User
Expert
Quality
control
System Testing
Integration, units, modules
Module
Module Test
Developer
Quality
Control
Experts
Deployment & Maintenance
installation
training
feedback
User
Installation
team
Enhancement Strategy
Closing
Analysis
Achieving
Learning
Upgrading
strategy
Project
team

Fig. 1. 4-Layed Plan-Driven Model (4-LPdM).

### G. Timeframe for a Phase

4-LPdM specified four major layers instead of phases of traditional methodologies due to incorporate project management capabilities; and each phase is specified with actions, process and respective stakeholders but not fixed the timeframe that depends on the size and complexity of the functionalities and resource availabilities. External stakeholders' requirements and business needs also influence the calculation of time. 4-LPdM suggests that a manager should consider additional time for meetings, communication, and analysis for each level besides resource specification. Moreover, it shows importance to implement project management tools and techniques to calculate, visualize, and manage schedule.

### H. 4-LPdM and Quality Control

4-LPdM is highly visible to present the task with the designed required stakeholder, tool, technique, and process that should be maintained by the execution team. The visibility features of the methodology help to determine the realistic time, cost, and resources to improve quality. Customer or user involvement significantly impacts the methodologies, and it specified the purpose of their interaction for a particular task or process. It allows customer interaction more than the heavy-weighted traditional methodologies and reduces over-interaction of light-weighted methodologies, so it ensures reasonable and justified customer interaction that improves customer satisfaction. The model proposed for specialized project management of near-future software development with artificial intelligence, smart infrastructure with IoT, and big data processing for knowledge retrieval. So, it explicitly specifies each task, process, and domain information on standard documentation (e.g., controlled language) to reduce ambiguity which enhances project management capability and quality.

Traditionally time, cost, and scope are considered the most significant influential factors of a software project to maintain the quality; that is updated by the project management body of knowledge (PMBOK 4.0) with six factors: scope, budget, quality, schedule, risk, resources [42] [44]. Mohammed et al [43] developed a six-pointed star model to evaluate the effectiveness of their proposed model with factors time, product (scope), risk, cost, and resource [43] [45]. Customers want to get a product on time within their budget and that should carry all functionalities so quality is described by customers' satisfaction and business goal of the organization. Fig. 2 is the quality model that consists of four factors to ensure the quality of the project based on the traditional model of scope, cost, and time. The project team considers a project as a successful project when the customer is satisfied with functional and non-functional requirements within schedule, budget, and scope. The task of a project is executed by a systematic process that should be well documented according to standardization guidelines. Effective resource allocation reduces extra cost, and an appropriate tool accomplishes a process on time without compromising the quality of the product. Stakeholders are guided by the model to practice formal documentation and responsible resource utilization that improve the efficiency of the workflow. The comprehensive and concrete quality control model is proposed (Fig. 2) to guide the execution of 4-LPdM. 4-LPdM is the main contribution of this research that will guide software practitioner to enhance the organizations' quality in software project management and improve quality of the product. This model inspires to practice guidelines of standardization organizations but not recommend for certification (individual choice). It is also suggested for standard organizations that has capability of utilizing tools and technologies.

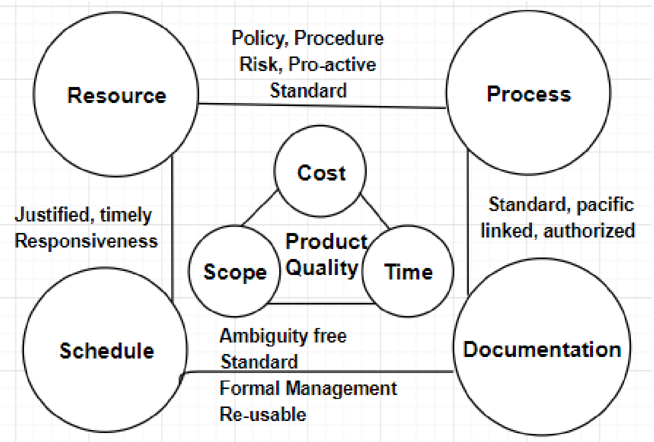

Fig. 2. Quality Control by 4-LPdM.

## IV. Research Methods

This section consists of data collection and analysis. Individual expert opinions are collected to determine the validity and efficiency of the proposed model. Furthermore, the proposed model is presented to the experts before data collection. Scope, cost, time, quality, sustainability, risk, resource, and management are considered influential factors according to the suggestion of experts of software developing companies. According to the project management body of knowledge (PMBOK 4.0), the advanced model of triple constraints consists of quality, scope, resource, budget, risk, and schedule known as influential factors of software project management. But management is one of the most vital factors to make a project successful; and sustainability improves quality by reducing wastage of resources: time, cost; and assets. So, management and sustainability are the additional two factors that are considered for the evaluation of this model. Software Requirement Specification (SRS) and milestone of a project are controlled by scope factors; a realistic plan is developed and executed with schedule factors; the budget parameters are justified by the return of the investment; resource factors are used to ensure efficient utilization of assets; sustainability factors improve the quality of process and product, while overall satisfaction is measured by the quality factors. Furthermore, complexity, understandability, and appropriateness are three more criteria that are considered for general reflection. Validity and efficiency checking of the proposed model is the main aim of the identification of the aforementioned factors.

For evaluation, a set of well-known software firms are invited from Bangladesh and abroad who have local and/or international experience. 29 representatives from 20 organizations were accepted to attend the evaluation process who were trained by poster and online presentation. They study the proposed methodology and try to implement their project (exiting / new project) for a month and evaluate based on the findings. Participants are selected from different levels of experience manager, developer, lead developer, software engineers, and system analysts, and more than 50% have multiple levels of experience including free lunching, individual, and teamwork.

A survey was conducted to collect feedback from experienced people of software firms. 29 respondents give feedback who are from 20 different organizations and 3 of them have freelancing experience. There were two different sections in the questionnaire: i) respondent and his/her organization's information was in section-1 and ii) section-2 carries responses for the proposed model. Table I (a) shows the respondents' experience in software production. Table I (b) describes the mostly practicing methodologies of the organizations. Survey respondents were related to all phases of the software development life cycle. They had different experiences on different types of projects. System analysts, designers, requirement engineers, developers, managers, testers, marketers are the common types of respondents. Respondents' current position in their organization is described in Table I (c). Table I (d) illustrates the respondent's professional experience.

TABLE I. RESPONDENTS' ANALYSIS

| **(a)** Production Classification | | |
|---|---|---|
| **Respondent's Experience** | **Frequency** | **Percentage** |
| Local Production | 10 | 34.48% |
| International Product | 11 | 37.93% |
| In house Product | 4 | 13.79% |
| Local & Global Product | 4 | 13.79% |
| | | |
| **(b)** Practicing Methodology | | |
| **Respondent's Experience** | **Frequency** | **Percentage** |
| Waterfall Methodology | 3 | 10.34% |
| Agile Methodologies | 15 | 51.72% |
| PRINCE2 Methodology | 2 | 6.90% |
| Self-developed Methodology | 7 | 24.14% |
| Other Methodologies | 2 | 6.90% |
| | | |
| **(c)** Respondent Position | | |
| **Respondent's Experience** | **Frequency** | **Percentage** |
| Developer | 5 | 17.24% |
| Lead Developer | 9 | 31.03% |
| Manager | 5 | 17.24% |
| System Analyst | 4 | 13.79% |
| Software Engineer | 6 | 20.69% |
| | | |
| **(d)** Respondents' Experience ( Years) | | |
| **Respondent's Experience** | **Frequency** | **Percentage** |
| 1-4 Years | 8 | 27.59% |
| 5-8 Years | 13 | 44.83% |
| 9-15 Years | 6 | 20.69% |
| More than 15 Years | 2 | 6.9% |

## V. EVALUATION

This section illustrates the statistical analyses of the collected numerical responses. These analyses aim to show the influence of each factor and how the management is related to each other. Table II shows the statistical analysis that compares eight factors in the form of Relatively-Importance. This methodology supports resource management (0.8344828) mostly and least interest in cost management (0.7609195). Table II is summarized from the data analysis of Appendix A. There is not much variation among the eight measuring factors (standard deviation). Appendix A describes the summarized result of collected responses that consists of all achieved frequency of 29 participants. Total frequency and computed percentage weight are represented according to the Likert scale. The significance of each factor of the proposed model is reflected in Appendix A, hence it shows that "strongly disagree" and "disagree" are too much less than comparatively "strongly agree" and "agree". Hence, only the "strongly agree", "agree", and "neutral" frequency table is illustrated in Fig. 3.

The average score for all factors is in between 3 and 4 of Likert scale (strongly agree=5, agree=4, neutral=3, disagree=2, strongly disagree=1). Fig. 3 represents responses for each sub-category (for example scope has three subcategories (A, B, C) and satisfactory is comparative more than any other options. The average score of all sub-categories is the final score for each category. 3.8 to 4.2 are the average score of the proposed model that is very near to agree (4) on the Likert scale. For example, the Likert scale value is multiplied with the average value of each factor then again calculate the average for all responses of this domain (scope, plan, etc.). Thus, average score represents the positive feedback in all aspects with minor variation. Standard deviation from 2.4 to 2.8 and according to the empirical role, it shows more than 95% response lies beside the means (i-1 to i+1). Relative importance is calculated ($R_i$=1/5$N_i$ (5$n_5$+4$n_4$ +3$n_3$+2$n_2$+1$n_1$)) and presented in Table II to show the importance of factors. The values are very close to each other, and the range is 0.7609195 to 0.8344828 indicates correlation.

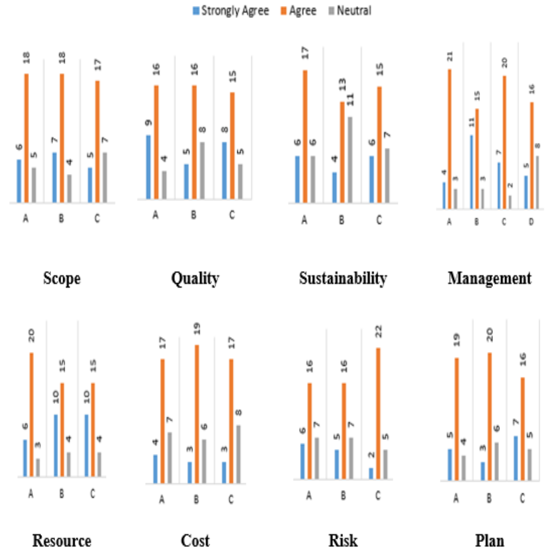

Fig. 3. Frequency Graph for All Measuring Factors.

TABLE II. STATISTICAL ANALYSIS OF THE 8 FACTORS

| Factor | Mean | Standard Deviation | Relatively Importance |
|---|---|---|---|
| Scope | 4.1944828 | 2.7620831 | 0.8045977 |
| Quality | 4.0344828 | 2.4832522 | 0.8068966 |
| Sustainability | 3.862069 | 2.410228 | 0.7724138 |
| Management | 4.0775862 | 2.8300027 | 0.8155172 |
| Resources | 4.1724138 | 2.699799 | 0.8344828 |
| Cost | 3.8045977 | 2.7849991 | 0.7609195 |
| Risk | 3.908046 | 2.8326754 | 0.7816092 |
| Plan | 3.954023 | 2.8490392 | 0.7908046 |

TABLE III. OVERALL SATISFACTION

| 4-LPdM Implementation | | 4-LPdM Understanding | | Appropriate for Projects | |
|---|---|---|---|---|---|
| Complex | 20.7% | Easy | 31% | Medium to large | 72.4% |
| As usual | 55.2% | Acceptable | 58.6% | Small to medium | 13.8% |
| Simple | 24.1% | Difficult | 10.4% | All | 13.8% |

Therefore, the relative importance ranks indicate positively correlated with other factors. Table III shows general aspects of three measures of complexity of the model, understanding the functionality of the model, and when it is suitable. The average percentage (55.2%) of respondents found that it is as usual as others in terms of complexity. 20.7% considered the presentation of the methodology is complex while 24.1% feels it is simple for them. The respondent understands from the presentation and poster and after implementation their project 10.4% feels difficult due to the explicit information specification. 31% feel easy to understand for implementation but more than 58% recommended as acceptable. This methodology is recommended for medium to large projects (72.4% in Table III) because of the extra activities for information specification that will accelerate cost for the small projects.

## A. Comprehensive Comparative Study

Waterfall involves users and customers only at the initial stage of the project, so it freezes requirements and documentation at the first phase. It also faces uncertainty problems and measuring the progress of the project is difficult too. The proposed model allows customer interaction and requirement flexibility until the design is finalized as well as risk and quality management mitigate uncertainty. The agile methodologies aim to accomplish a project in a short time that could compromise with quality, and lack of documentation practice makes problems in the re-usability of design and code. Furthermore, new employees struggle in an agile team for technology transfer and highly functional dependency projects. A well-structured and documentation practice of the proposed model demolishes the issues of agile methodologies. Spiral is good for high-risk, complex, and without time constraint projects but time and budget are crucial factors of any project, but the proposed model is suitable for medium to large projects with time constraints. An iterative approach does not fix requirements at the early stage that may cause ambiguous requirement specifications, it allows more customer interaction and informal practice that could make problems to accomplish a project in time and budget. The proposed model allows user interaction only before fixing the design. In addition, its' formal practice of communication and documentation addresses the limitation of the iterative model.

4-LPdM adopts the plan-driven because more information specification is required for the smart information system, artificial intelligence applications, and big data analytics. Descriptive logic is proposed for documentation due to ambiguity reduction and demolish language barriers in distributed project management; moreover, it can easily convert to computer language. Sustainability is focused to ensure the re-usability of documentation and soft resources for a project to another project. The project management approach is integrated with the system development life cycle to improve the management process, risk reduction, and maximize resource utilization. It is validated by common six factors but also considered other special challenges that appear for new technologies like cloud computing, mobile applications, IoT, AI, and big data project. Table IV describes the role of the proposed methodology for IoT, AI, data science, and distributed projects for current days.

TABLE IV. OVERALL SATISFACTION

| Project | Special Features | Role of 4-LPdM |
|---|---|---|
| IoT | Connectivity, sensing, scale, dynamic, intelligence, energy, safety, integration [47]. | IoT devices should be active and act according to the outcome of intelligence applications. Edge, fog, or cloud computing perform sensing data processing where data processing and sharing is an important activity. 4-LPdM specifies data design (Fig. 1) and suggests formal documentation (Fig. 2) to minimize risk. |
| AI | Complex training algorithms, computation power, data selection and security, integration, and infrastructure [48]. | 4-LPdM formal analysis, documentation, and review for each layer; minimize complexity and help to select appropriate algorithms; improve data quality and security; support data interrogation; and manage with a suitable infrastructure. It suggests explicit data specification and design that can improve logic, axioms, grammars, and role fixing for AI applications. |
| Data Science | Weakness in data literacy and culture, measuring of data science project, stakeholders' cooperation, and trust on solution [39][49]. | Stakeholders' cooperation is one of the most important factors in data science to achieve business goals, and (Fig. 1) the proposed methodology includes shareholders in the quality frame (Fig. 2). It also allows explicit data specification [46] and communication roles in the planning phase. It integrates capacity building factors, staff training facilities, measuring tools and techniques, customer interactions, and feedback-accepting systems (Planning phase of Fig. 1) to improve trust. |
| Distributed | Communication, culture, ownership, misunderstanding, knowledge transfer woes, and hassle of integration [12]. | Distributed project management system is new and becoming popular to access low wage technical employees from another part of the world or practicing procurement management for a part of the project. 4-LPdM has a plan-driven approach and recommends using control languages to reduce misunderstanding, support knowledge transfer, and improve communication. |
| Mobile apps | More interactive with usability and accessibility functionalities [52]. | Proposed system will support for interactive, usability, and accessibility features specification in system design phases that will be usable for further similar type projects. |
| Cloud projects | Public and hybrid clouds project faces migration challenges [51]. | Cloud infrastructure become popular due to the pay as you model and small companies migrated to cloud platform. If an on-premises project is completed with 4-LPdM it will be easily migrated to public cloud due to the formal documentation and explicitly project data specification. |

## VI. CONCLUSION AND LIMITATIONS

A quality project management approach is extremely important to accomplish a successful software project within a predefined time and budget. Furthermore, a product's quality depends on the quality of the process, tasks, and stakeholders that is guided by a methodology. The literature review reveals the importance of mitigating the existing limitations and gaps in software methodologies. Moreover, a methodology should be adaptable and predictable with people, tasks, and processes. The proposed methodology is going to fill up the gap and reduce the limitations by introducing a concrete framework of micromanagement architecture, project management approach, and system development phases. It will enhance the managerial capability by formal process and practice with maximizing stakeholders' responsibility. A quality control unit is adjusted with each stage of the project that will ensure quality and minimize risk.

This model is developed and evaluated according to the opinion of experts and the survey result positively indicates the importance of the proposed model. Statistical analyses (means, standard deviation, relative importance) are applied for scoring the result and positive feedback is reflected in all factors. Therefore, it is appropriate for any standard software developing company.

This is a simplified model that separates virtual management functionalities from the traditional approaches. It overlooked the explanation of traditional phases that will enable an adaptation of existing traditional approaches. The standard software firms that have specific business goals can access talent from any corner of the world. It avoids the complexity of a virtual project management. An ad-hoc or special software could be managed by online procurement management, but it is not logical for a standard organization to host everything in outsourcing without proper utilization of organizational resources. Management software is 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 a user can attend testing in distance mode. While the integrated system is considered into the organization (physical mode). In the future, this work will be extended for a fully virtual mode project management approach.

Limitations: 4-LPdM is a common methodology that is proposed for any software project but considered to resolve latest issues (Section II) too. So, it is evaluated by different types of practitioners (Table I) and based on the six common criteria of general perspective (Table II) to show the overall acceptance for any software project. Number of participants and duration of practice could be increased for further study. It could be extended to the specific software project of AI, IoT, big data and evaluated by the respective experts. Moreover, distributed, or virtual project management approach can implement and evaluate too.

## ACKNOWLEDGMENT

The proposed models are evaluated by a group of experts during June-August 2020. We are thankful to the valuable experts who have distinguished experiences in the software industry. Their support is appreciable and makes our research successful. This data is not used for any commercial purposes.

## REFERENCES

[1] Henseler, Tomás. (2018). Infographic: Timeline of Software Development Methodologies. Retrieved January 15, 2020, from https://www.hexacta.com/timeline-of-software-development-methodologies/.

[2] P. Nistala, K. V. Nori and R. Reddy, "Software Quality Models: A Systematic Mapping Study," 2019 IEEE/ACM International Conference on Software and System Processes (ICSSP), Montreal, QC, Canada, 2019, pp. 125-134, doi: 10.1109/ICSSP.2019.00025.

[3] AL-Badareen, Anas & Selamat, Mohd & A. Jabar, Marzanah & Din, Jamilah & Turaev, Sherzod. (2011). Software Quality Models: A Comparative Study. Communications in Computer and Information Science. 179. 46-55. 10.1007/978-3-642-22170-5_4.

[4] Standards Organizations. (n.d.). Retrieved March 12, 2020, from https://www.oss.com /resources/standards-organizations.html?gclid=Cj0KCQjwoPL2BRDxA RIsAEMm9y 83GhM3-jQEckajEs1eBDs2aycJqsIt_RcCinfyDbnYypoEwUL7KiQaAuTaEALw_wcB.

[5] The Standish Group report 83.9% of IT projects partially or completely fail. (n.d.). Retrieved April 3, 2020, from https://www.opendoorerp.com/ the-standish-group -report-83-9-of-it -projects- partially-or-completely-fail/.

[6] Bent Flyvbjerg and Alexander Budzier (2011). Why Your IT Project May Be Riskier Than You Think. available: https://hbr.org/2011/09/why-your-it-project-may-be-riskier-than-you-think#comment-section, visited 18th nov 2019.

[7] PMI, 2017. Success Rate Rise, Performing the high cost of low performance. Available:https://www.pmi.org/-/media/pmi/documents/ public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2017.pdf visited: 18th Nov 2019.

[8] Geneca, 2017.Why up to 75% of the project will fail? Available:https://www.geneca.com/why-up-to-75-of-software-projects-will-fail/ visited:19 Nov 2019.

[9] Horvath, K. (2018, October 29). How Digital Transformation Impacts Software Development. Retrieved April 3, 2020, from https://content.intland.com/blog/how-digital-transformation-impacts-software-development.

[10] CIKLUM. (2019, September 28). 7 Evolving Trends in Software Development. Retrieved May 7, 2020, from https://www.ciklum.com/blog/7-evolving-trends-in-software-development/.

[11] M. Broy, "Yesterday, Today, and Tomorrow: 50 Years of Software Engineering," in IEEE Software, vol. 35, no. 5, pp. 38-43, September/October 2018, doi: 10.1109/MS.2018.290111138.

[12] ReQtest. (2019, August 6). 7 Project Management Challenges in Distributed Development. Retrieved May 6, 2020, from https://reqtest.com/development/project-management-challenges-distributed-development/.

[13] Cândido, C. J., & Santos, S. P. (2015). Strategy implementation: What is the failure rate?. Journal of Management & Organization, 21(2), 237-262.

[14] Kerzner, H., & Kerzner, H. R. (2017). Project management: a systems approach to planning, scheduling, and controlling. John Wiley & Sons.

[15] Davis, K. (2014). Different stakeholder groups and their perceptions of project success. International journal of project management, 32(2), 189-201.

[16] Kerzner, H., & Kerzner, H. R. (2017). Project management: a systems approach to planning, scheduling, and controlling. John Wiley & Sons.

[17] Discenza, R. & Forman, J. B. (2007). Seven causes of project failure: how to recognize them and how to initiate project recovery. Paper presented at PMI® Global Congress 2007—North America, Atlanta, GA. Newtown Square, PA: Project Management Institute.

[18] Kulish, A. (2019, November 20). Why Quality Software Is Impossible Without Proper Root Cause Analysis (RCA). Retrieved April 20, 2020, from https://www.infopulse.com/blog/why-quality-software-is-impossible-without-proper-root-cause-analysis-rca/.

[19] A Mandal and S. C. Pal. Identifying the Reasons for Software Project Failure and Some of their Proposed Remedial through BRIDGE Process Models.International Journal of Computer Sciences and Engineering. Vol.-3(1), PP(118-126) Feb 2015, E-ISSN: 2347-2693.

[20] Elzamly, A., Hussin, B., & Salleh, N. (2015). Methodologies and Techniques in Software Risk Management Approach for Mitigating Software Failure Risks: A Review. Asian Journal of Mathematics and Computer Research, 2(4), 184-198. Retrieved from http://www.ikprress.org/index.php/AJOMCOR/article/view/63.

[21] A. R. Gilal, J. Jaafar, S. Basri, M. Omar and A. Abro, "Impact of software team composition methodology on the personality preferences of Malaysian students," 2016 3rd International Conference on Computer and Information Sciences (ICCOINS), Kuala Lumpur, 2016, pp. 454-458, doi: 10.1109/ICCOINS.2016.7783258.

[22] Geamba, Cristina & Jianu, Iulia & Jianu, Ionel & Gavrila, Alexandru. (2011). Influence Factors for the Choice of a Software Development Methodology. Journal of Accounting and Management Information Systems. 10. 479-494.

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

[24] Adenowo, Adetokunbo A. A. and Basirat A. Adenowo. “Software Engineering Methodologies: A Review of the Waterfall Model and Object-Oriented Approach.” (2013).

[25] 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.

[26] Dmitry Gurendo. "Software Development Life Cycle (SDLC): Spiral Model". XB Software. October, 2015. Retrieved on 20th November 2020 from https://xbsoftware.com/blog/software-development-life-cycle-spiral-model/.

[27] S. Mathur and S. Malik, ‘‘Advancements in the V-model,’’ Int. J. Comput. Appl., vol. 1, no. 12, pg. 29-34, 2010, doi: 10.5120/266-425.

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

[29] Birkinshaw, J. (2018). What To expect from agile. MIT Sloan Management Review, 39–42. https://doi.org/10.1142/S201000781100019X.

[30] Baham, C., Hirschheim, R., Calderon, A. A., & Kisekka, V. (2017). An agile methodology for the disaster recovery of Information Systems under catastrophic scenarios. Journal of Management Information Systems, 34(3), 633–663. https://doi.org/10.1080/07421222.2017.1372996.

[31] Varajão, J. E. (2018). A new process for success management. Journal of Modern Project Management, 92–99. https://doi.org/10.1177/004051755602600608.

[32] Stoica, M., Ghilic-Micu, B., Mircea, M., & Uscatu, C. (2016). Analyzing agile development – from waterfall style to Scrumban. Informatica Economică, 20(4), 5–15.

[33] Rajagopalan, S., & Mathew, S. K. (2016). Choice of agile methodologies in software development : A vendor perspective. Journal of International Technology and Information Management, 25(1).

[34] Anderson, D.J. 2010. Kanban: Successful Evolutionary Change for Your Technology Business, Blue Hole Press, Ed.1.

[35] Fahad, Muhammad & Qadri, Salman & Ullah, Dr. Saleem & Husnain, Mujtaba & Qaiser, Rizwan & Qureshi, Shehzad & Ahmed, Waqas & Syed, Shah & Muhammad, Syed. (2017). A Comparative Analysis of DXPRUM and DSDM. International Journal of Computer Science and Network Security, VOL.17 No.5, May 2017.

[36] Jugdev, K., Perkins, D., Fortune, J., White, D. and Walker, D. (2013), "An exploratory study of project success with tools, software and methods", International Journal of Managing Projects in Business, Vol. 6 No. 3, pp. 534-551. https://doi.org/10.1108/IJMPB-08-2012-0051.

[37] Walker, D. and Lloyd-Walker, B. (2019), "The future of the management of projects in the 2030s", International Journal of Managing Projects in Business, Vol. 12 No. 2, pp. 242-266. https://doi.org/10.1108/IJMPB-02-2018-0034.

[38] K. U. Sarker, A. Bin Deraman, R. Hasan, S. Mahmood, A. Abbas and M. Sohail, "Kids’ Smart Campus Ontology to Retrieve Interest," 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC), Muscat, Oman, 2019, pp. 1-4. doi: 10.1109/ICBDSC.2019.8645585 available: https://ieeexplore.ieee.org/document/8645585.

[39] Sarker, Kamal Uddin; Deraman, Aziz Bin; Hasan, Raza; Abbas, Ali. Ontological Practice for Big Data Management. International Journal of Computing and Digital Systems. Volume-8, issue-3. Pp:265-272

May2019. DOI: 10.12785/ijcds/080306. Available: https://journal.uob.edu.bh/handle/123456789/3485.

[40] K. U. Sarker, A. B. Deraman and R. Hasan, "Descriptive Logic for Software Engineering Ontology: Aspect Software Quality Control," 2018 4th International Conference on Computer and Information Sciences (ICCOINS), Kuala Lumpur, 2018, pp.1-5. doi: 10.1109/ICCOINS.2018.8510585 available: https://ieeexplore.ieee.org/document/8510585.

[41] Kamal Uddin Sarker, Dr. Aziz Deraman, Raza Hasan, Abdul Hadi Bhatti, "Software Development Life Cycle Developing Framework" 5th International Conference on Communication and Computer Engineering (ICOCOE 2018) Melaka, Malaysia. https://maltesas.my/msys/explore/pubs/1562833286/1562833286_Article_1564562131.pdf.

[42] K. Schwaber, Agile Project Management With Scrum. Redmond, WA, USA: Microsoft Press, 2004.

[43] MUHAMMAD AZEEM AKBAR, JUN SANG, ARIF ALI KHAN3, FAZAL-E-AMIN4, NASRULLAH, MUHAMMAD SHAFIQ, SHAHID HUSSAIN, HAIBO HU, MANZOOR ELAHI, AND HONG XIANG, "Improving the Quality of Software Development Process by Introducing a New Methodology—AZ-Model". IEEE Access. VOLUME 6, pp.4811-4823,2018, DOI: 10.1109/ACCESS.2017.2787981.

[44] Kamal Uddin Sarker, Aziz Deraman, Raza Hasan, Ali Abbas, Babar Shah, Abrar Ullah. Monolithic Ontological Methodology (MOM): An Effective Software Project Management Approach" Journal of Engineering Research. Vol 10, No 2A, 2022.

[45] Kamal Uddin Sarker, Aziz Bin Deraman, Raza Hasan and Ali Abbas, "SQ-Framework for Improving Sustainability and Quality into Software Product and Process" International Journal of Advanced Computer Science and Applications(IJACSA), issue 9, volume 11, page 69-78. ISSN : 2156-5570 (Online) ISSN : 2158-107X (Print) 2020. http://dx.doi.org/10.14569/ IJACSA.2020.0110909.

[46] Sarker KU, Deraman A, Hasan R, Abbas A. Explicit specification framework to manage software project effectively. Indian Journal of Science and Technology. volume 13, issue 36, page 3785-3800, year 2020. https://doi.org/10.17485/IJST/v13i36.1244 https://indjst.org/articles/explicit-specification-framework-to-manage-software-project-effectively.

[47] Priya Pedamkar. "IoT Features". EDUCBA. 2022. Retrieved on 14th May 2022 from https://www.educba.com/iot-features/.

[48] 10xDS Team. "5 Common Challenges in Implementing Artificial Intelligence (AI)". 10Xds. March 2021. Retrieved on 14th May 2022 from https://10xds.com/blog/challenges-implementing-artificial-intelligence/.

[49] Gramener Inc. "Challenges in Data Science Projects and How to Tackle Them". September 2020. Retrieved on 15th May 2022 from https://blog.gramener.com/data-science-project-challenges/.

[50] Megan Keup. "Virtual Project Management: Benefits, Challenges & Tools". April 2020. Retrieved on 2nd June 2022 from https://www.projectmanager.com/blog/virtual-project-management.

[51] Rushi Patel. 10 Biggest Cloud Computing Challenges in 2022 for IT Service Providers. April 2022. Retrieved on 6th June 2022 from https://www.mindinventory.com/blog/cloud-computing-challenges/.

[52] Radosław Szeja. 14 Biggest Challenges in Mobile App Development in 2022. Netguru. January, 2022 .Retreived on 1st June 2022 from https://www.netguru.com/blog/mobile-app-challenges.

## APPENDIX A

### Practitioners' Data Analysis

It consists of the analysis of responses of all individual question. There are six major criteria validated by practitioners and each of them divided into 3 to 4 sub criteria to get more accurate reflection. Each question is having criteria for the proposed methodology. For example, the score of resource is the average score of A. how much supportive for hardware/software resource utilization, B. how much suggestive for proper utilization of human resource, and C. how much guided for resource sharing and responsibly handling? There are 29 participants and 6, 20, and 3 are the responses for strongly agree, agree, and neutral; and their percentage are respectively 20.69, 68.97, and 10.34.

TABLE V. OVERALL SATISFACTION

| Factor | Questions | Strongly Agree (Response and %) | Agree (Response and %) | Neutral (Response and %) | Disagree (Response and %) | Strongly Disagree (Response and %) | Total and % |
|---|---|---|---|---|---|---|---|
| Scope | A. The proposed framework will guide to clarify the scope of the project | 6 | 18 | 5 | 0 | 0 | 29 |
| | | 20.69 | 62.07 | 17.24 | 0 | 0 | 100 |
| | B. It will help to monitor the scope of the project | 7 | 18 | 4 | 0 | 0 | 29 |
| | | 24.14 | 62.07 | 13.79 | 0 | 0 | 100 |
| | C. It will help to meet the scope | 5 | 17 | 7 | 0 | 0 | 29 |
| | | 25 | 85 | 35 | 0 | 0 | 100 |
| Quality | A. The proposed framework will help to quality product | 9 | 16 | 4 | 0 | 0 | 29 |
| | | 31.03 | 55.17 | 13.79 | 0 | 0 | 100 |
| | B. It will improve client satisfaction | 5 | 16 | 8 | 0 | 0 | 29 |
| | | 17.24 | 55.17 | 27.59 | 0 | 0 | 100 |
| | C. It will suggest a quality working environment | 8 | 15 | 5 | 1 | 0 | 29 |
| | | 27.59 | 51.72 | 17.24 | 3.448 | 0 | 100 |
| Sustainability | A. The proposed framework will improve economic sustainability | 6 | 17 | 6 | 0 | 0 | 29 |
| | | 20.69 | 58.62 | 20.69 | 0 | 0 | 100 |
| | B. It will improve social sustainability | 4 | 13 | 11 | 1 | 0 | 29 |
| | | 13.79 | 44.83 | 37.93 | 3.448 | 0 | 100 |
| | C. It will improve environment sustainability | 6 | 15 | 7 | 1 | 0 | 29 |
| | | 20.69 | 51.72 | 24.14 | 3.448 | 0 | 100 |
| Management | A. The proposed framework will improve formal management | 4 | 21 | 3 | 1 | 0 | 29 |
| | | 13.79 | 72.41 | 10.34 | 3.448 | 0 | 100 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | B. It will improve professionalism | 11 | 15 | 3 | 0 | 0 | 29 |
| | | 37.93 | 51.72 | 10.34 | 0 | 0 | 100 |
| | C. It will help to monitor and control | 7 | 20 | 2 | 0 | 0 | 29 |
| | | 24.14 | 68.97 | 6.897 | 0 | 0 | 100 |
| | D. It will help to distribute project management | 5 | 16 | 8 | 0 | 0 | 29 |
| | | 17.24 | 55.17 | 27.59 | 0 | 0 | 100 |
| **Response** | A. The proposed framework will enhance material resource utilization | 6 | 20 | 3 | 0 | 0 | 29 |
| | | 20.69 | 68.97 | 10.34 | 0 | 0 | 100 |
| | B. It will enhance human resource management | 10 | 15 | 4 | 0 | 0 | 29 |
| | | 34.48 | 51.72 | 13.79 | 0 | 0 | 100 |
| | C. It will improve cooperation and responsiveness | 10 | 15 | 4 | 0 | 0 | 29 |
| | | 34.48 | 51.72 | 13.79 | 0 | 0 | 100 |
| **Cost** | A. The proposed framework will guide to identify cost factors | 4 | 17 | 7 | 1 | 0 | 29 |
| | | 13.79 | 58.62 | 24.14 | 3.448 | 0 | 100 |
| | B. It will guide to cost control | 3 | 19 | 6 | 1 | 0 | 29 |
| | | 10.34 | 65.52 | 20.69 | 3.448 | 0 | 100 |
| | C. It will help to complete the project on budget | 3 | 17 | 8 | 1 | 0 | 29 |
| | | 10.34 | 58.62 | 27.59 | 3.448 | 0 | 100 |
| **Risk** | A. The proposed framework will help to avoid the risk | 6 | 16 | 7 | 0 | 0 | 29 |
| | | 20.69 | 55.17 | 24.14 | 0 | 0 | 100 |
| | B. It will help to make the risk mitigation plan | 5 | 16 | 7 | 1 | 0 | 29 |
| | | 17.24 | 55.17 | 24.14 | 3.448 | 0 | 100 |
| | C. It will focus on business objectives to meet the risk | 2 | 22 | 5 | 0 | 0 | 29 |
| | | 6.897 | 75.86 | 17.24 | 0 | 0 | 100 |
| **Plan** | A. The proposed framework will guide to develop a realistic plan | 5 | 19 | 4 | 1 | 0 | 29 |
| | | 17.24 | 65.52 | 13.79 | 3.448 | 0 | 100 |
| | B. It will help to execute according to plan | 3 | 20 | 6 | 0 | 0 | 29 |
| | | 10.34 | 68.97 | 20.69 | 0 | 0 | 100 |
| | C. It will help to accomplish on time | 7 | 16 | 5 | 1 | 0 | 29 |
| | | 24.14 | 55.17 | 17.24 | 3.448 | 0 | 100 |