IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 7, 2021, accepted May 7, 2021, date of publication May 11, 2021, date of current version May 20, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3079194

Green-Agile Maturity Model: An Evaluation Framework for Global Software Development Vendors

NASIR RASHID[1], **SIFFAT ULLAH KHAN**[1], **HABIB ULLAH KHAN**[2], **AND MUHAMMAD ILYAS**[1]

[1]Department of Computer Science and IT, University of Malakand, Chakdara 18800, Pakistan
[2]Department of Accounting and Information Systems, College of Business and Economics, Qatar University, Doha 2713, Qatar

Corresponding author: Habibullah Khan (habib.khan@qu.edu.qa)

This work was supported by the Qatar National Library, Doha, Qatar, and in part by the QU Internal Grant Qatar University Internal under Grant IRCC-2021-010.

ABSTRACT Agile methods are extensively adapted by software development organizations due to the competitive benefits it offers. In recent years global software development (GSD) projects practice agile methods as prominent methods to deliver the software in increments with utmost user satisfaction and affordable cost. Beside the use of agile methods, the software industry has also considered the green aspect of software, to be in line with the demands of the organizations and the world technological ecosystem. The green and sustainable feature of software should focus both the energy and resource efficiency key factors. This phenomenon of embedding the green flavor in software has emerged a new research area, green software engineering, that promises the development of eco-friendly software with minimum energy and use of less computing resources, to trim down the adverse effects on both society and environment. The principal objective of this research study is to design and develop a multi-level Green-Agile Maturity Model (GAMM) to assess the GSD vendors' agile maturity in terms of green software development. The model has been built in four phases. In phase I and II, systematic literature review (SLR) was performed to identify the success factors and risk factors that either supports or hinders the green and sustainable software development respectively by practicing the agile methods in GSD. The results have been validated from 106 relevant experts, dealing with agile and green software projects, through questionnaire survey. The experts' demographic represents 25 different countries. We also identified the industry practices through SLR and survey, to address our identified critical factors. Phase III of this research deals with development of the GAMM by categorizing the identified factors into seven Green-Agile maturity levels. A similar approach has been used in other models such as Capability Maturity Model Integration (CMMI), Implementation Maturity Model (IMM) and Software Outsourcing Vendors Readiness Model (SOVRM). In phase IV of this research, five case studies were conducted at GSD organizations, to evaluate the structure and efficacy of the GAMM, while as a major contribution, this paper presents our developed model, the GAMM, which aims to assess the green-agile maturity of the GSD vendors in terms of green and sustainable software development.

INDEX TERMS Agile software development, green and sustainable software, green-agile maturity model (GAMM), global software development (GSD), agile methods.

I. INTRODUCTION

Agile software development (ASD) emphasis on customer satisfaction to meet their functional requirements with realistic development schedule and to deliver the actual software code rapidly. Unlike conventional development, agile software development get credit in scaling down the risks and multiply the software productivity [1]. Agile methods key factor is to trust the developers' skills and their in-time development to deliver the functional code rather than producing than adapting lengthy and formal procedures with substantial documentation, as documentation is a poor form of communication, but sometimes it is obligatory to retain critical information over time [2]. Agile methods strive to

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

avoid formalized and time intensive processes that actually do not contribute in delivering the functional code of software [3].

Agile Alliance laid a platform to formally publish the agile manifesto [4] which contributed the agile methods to faster the software development processes and deliver the working code in minimum time. The agile manifesto introduced a new paradigm with industry-led vision for reflective change in software development. Agile methods stress on customer satisfaction with accepting frequent changes, faster delivery of working software code, simple design and interface, and rich communication with customers [5], [6].

With an increasing demand for more complex software applications, Information and Communication Technologies (ICTs) bear a huge negative impact on the environment due to its increased utilization of resource and power consumption. Literature reveals that the effect of ICTs on sustainable development specifically on software is the hot topic in the area of Green Computing [7], [8].

''Green or sustainable software development is the design and production of software, having direct or indirect negative effect on country's economy, people, society and environment that result from software pre-development, development and post-development phases are negligible and/or which have a positive impact on sustainable production'' [9].

''Green Software Engineering (GSE) is the art of developing green and sustainable software with a green and sustainable software engineering process'' [10]. GSE is a flourishing and assertive paradigm shift with exponential growth in software engineering. Green software claims to produce software with eco-friendly features to trim down the environmental effects triggered by software development. It promises to bolster the development of green and sustainable software in different phases of development and scale down its adverse effects on economy, society and environment [11].

Researchers have been tried to develop green software with sustainable processes to provide guidance and assist the stakeholders to focus on the new demanding trend for designing software products. Some researchers have focused to develop such applications that could assess the effectiveness of software development process and its use on the environment in terms of energy efficiency and resources utilization. Some reported research work focus on the design of operating system to control the usage of applications in terms of power consumptions [12], [13]. GSE has captivated the potentials of software developers with prime principals of developing the software with low minimum impact on society, economy, and ecology [14]. GSE targets to achieve software development with remarkable reduction of energy and natural resources and further to trim down the negative effects on human beings due to software development processes and its use [15].

Better solutions to meet the requirements for greener software processes [16], [17] include virtualization, closing applications that are no longer in use, efficient algorithms by writing a compact design of code and data structures, lessening parallelism overhead through effective algorithms, green computing, and designing well efficient energy allocation algorithms.

To cope the issues in developing green and sustainable software, integration of green aspects with agile practices is direly needed for entire customer satisfaction to deliver green and environment friendly software with reduced cost and minimal developmental resources [18].

As demand and need of the current era, green software [19]–[21] is a buzz word with phenomenal growth, to deliver green software. Inclusion of some effective agile principles in basic software engineering processes can result in environment friendly software [19]. Several practices of the ductile methods of agile software development have been adopted to achieve green and sustainable software [22]. Refining the software product continuously by accepting the real time requirements, in-time delivery of the functional code, focus to design and develop only what the customer need, and early detection of code errors are some of the principal features of agile methods that could add green aspect in software development [18], [23].

Our research is focused on green agility in the context of Global software development (GSD). GSD has gained massive success as it is compatible and acceptable global and interactive development paradigm in current software business. GSD facilitates world-wide software engineers with diverse cultures and distinct time zones, to play an active role in overall management and software development process. For consistent and effective collaboration, the experts use modern communication tools to share their knowledge and integrate the developed software modules [24]. GSD has several different paradigms. These include outsourcing (offshore, near-shore and on-shore), subsidiary development, freelancing, etc. [25]. GSD paradigms contribute numerous benefits to global software community. Some of the major benefits include availability of extensive developers round-the-clock with ease of access, to develop software with globally accepted standards, market accessibility for increased business, software updates with low affordability and adapting ''follow-the-sun'' development pattern [26]. Hence, software development can take place anytime, anywhere and is considered as global distributed projects [27]. GSD, as used in this research, refers to software development activity that involves two or more companies across the globe that combine their competencies and technologies to create new shared value while, at the same time, managing their respective costs and risks.

To reap the benefits of agile methods in the context of GSD, the integrated approach known as Agile Global Software Development, is in practice. The latest published reports stress the importance of agile practices in software development and emphasis to scale it to large development teams. Some studies [27], [28] demonstrate the exponential adaption of agile methods to grasp its benefits in GSD as well, such as faster delivery of high quality functional code, nominal cost and flexibility to accomplish the frequent changing requirements at every stage of development [29], [30].

The general objective of this research was to identify the critical success factors and risk factors, and to validate those factors through online questionnaire survey. From this general objective, we derived our specific objective: (i) to develop a Green-Agile Maturity Model (GAMM), based on the inputs from RQ1-RQ3 (ii) to conduct case studies at different agile GSD industry to evaluate the capability of the GAMM. The GAMM will assist the agile GSD vendors to quantify their maturity in developing green and sustainable software, using agile methods. After an in-depth analysis of the different models, very few works focus on maturity of software organizations but still there is a lack of such maturity model, to analyze the capability of vendor organization in developing green software, using agile methods evince the need of and confirms the novelty of our research work by contributing the GAMM. The main advantage of this model for the GSD organizations is that they could easily measure their green-agile maturity and could know about their strengths and weaknesses. Further, the GSD organization could compete the current software market, which demands green and sustainable software with increased adaption of agile methods, by addressing its core weakness, identified by the GAMM.

The rest of the paper is structured as follow: Section II presents the background study related to the research work. Section III manifests the prime research questions. Section IV describes the research methodology adapted to conduct the different phases of this research. Section V demonstrates the different steps involved in development of the GAMM. Section VI describes how the GAMM is evaluated by reporting the five case studies, conducted at different GSD organizations. Section VII interprets the feedback received from the case study participants about the basic criteria of the GAMM and how they evaluated it. Section VIII uncovers the limitations of research design. Section IX demonstrates the GAMM as a key contribution to the knowledge domain of software engineering in general and to GSD and green-agile community in specific, while Section X presents the conclusion and directions for future work.

II. BACKGROUND OF THE STUDY

Agile methods are extensively adapted by software community to meet the current needs of software development. ASD aims to change the development paradigm with continued efforts to overhaul the integral processes to ensure the release of green software through short increments [31]. ASD consists of customer-centric principles such as strong and productive team work, scheduled meetings between development team and business organizations, frequent discussions with active customers to finalize and refine the increments, early consignment of the functional code, and high-level flexibility to accommodate dynamic requirements [3], [32].

Mahmoud and Ahmad [19] have insisted on the use of agile principles in software development life cycle, to develop green software. To integrate some of the agile principles with software engineering processes can obviously aid in having environment friendly processes. Out of the twelve agile principles, such as, to embrace changes even late in software development, combined work strategy of developers and customers, incremental early development, iterative development and regular testing and evaluation can explicitly aid in energy efficient, green and sustainable software development.

The core agile principles such as, faster development and delivery in increments with shorter iterations, small builds with just enough design, and continuous code and increments integrations could support development of green software, used in military avionics systems [33].

Tate [22] asserted the delivery of quality software with embedded green features through agile methods with trained agile mind-set. Sustainable refinement of software with accommodating the frequent users' requirements, faster delivery of the working code, stress on design what is required and keeping it simple, and auto prevention of defects are some of the practical principals of agile methods that contribute in software sustainability [18], [23].

Incremental integration, high response to changing requirements, iterative development, continuous validation and frequent deliveries are the core principles of agile methods that can be beneficial for GSD and can eradicate the risks due to distributed environment [34].

Ray *et al.* [31] advised some valuable changes such as, reducing the carbon emission, power, and paper use for achieving greener software in the existing software development life cycle (SDLC). The projected model integrates sustainability reviews and journal with sustainability retrospective at various phases of SDLC to develop energy efficient software with minimal computing resources.

Dick *et al.* [20] contributed a model that emphasis on integrating agile practices with Green IT factors for software sustainability. The model has based a Scrum and has modified it through environment friendly practices. However, it lacks in providing a module to precisely measure the energy efficiency and carbon emission.

Abdullah *et al.* [23] shed light on the importance of knowledge management for the development of sustainable software. The authors used SLR as a research methodology to assess the available literature on GSD regarding the evolution of green computing and discuss how knowledge management comes to assist the management of GSD projects. The authors have discussed the importance of agile methods in terms of knowledge management for green and sustainable software development. According to the literature retrieved, applying the agile development processes in GSD projects would better manage the tacit knowledge for developing green software.

Kern *et al.* [35] have presented a generic model with supportive nature of software processes, to develop green and sustainable software. As the model is generic, so could be implemented in addition to the SDLC for energy efficient and environment friendly software design. The authors demonstrate an integrated model with the Scrum and called it agile SDLC to better assist the green software development processes. The proposed model incorporates few appropriate principles of Scrum, such as assessment of processes, reviewing sustainability in each phase, and documenting sustainability features, and software release in increments for in time delivery of sustainable software.

Pankowska [36] presents a distinct approach to sustainable software development, as well as a different perception of sustainability in software engineering. The author conducted a survey for the validation of practices and software development methods that are considered important for sustainable software product development. Particularly, the author has focused on agile methods in the context of human efforts as well as energy and computer power to support the sustainable software design and development. According to the survey conducted by the author, following are the crucial principles of agile methods that can be applied to achieve sustainable software development.

- Continual refinement of the product and project practices.
- Focus on delivery of the working product all the times.
- Continuous emphasis on simple software design
- Defect detection and prevention.

Stammel *et al.* [37] in his research investigation regarding the impact of agile methods on sustainable software development argue that agile software methodologies provide only a few implementation details of the software development process. This ‘‘low-level’’ approach may build the software that meet short-term individual project needs, but that do not necessarily lead to sustainable software. Their findings reveal that a lack of or insufficient system documentation may be a big hindrance to achieve sustainability in software systems and leads to increased system complexity, degraded maintainability, and lack of system familiarity.

Hsieh and Chen [38] discovered some patterns with high integration of increments in variable software development platforms. The authors documented that cross-platform software development faces a critical challenge in the context of practicing agile methods, i.e., a lack of direct involvement of customer in development life cycle. In absence of the customer, the software builds take a longer time to complete and are more likely to fail, thus have a negative impact on sustainable software development.

Taina [39] explains the various ways in which agile methodologies affect the sustainable software development. Among the different factors efficient time management and usage of computing resources are considered intrinsic factors that support the development of green software. However, encountering the risks’ management overhead in practicing agile methods may result in over budgeted projects, time over run, and eventually the maximum usage of computing resources, which could badly effect the sustainable processes for developing green software [40], [41].

According to Jan [42] green and sustainable software is the need of today and should be designed for entire customer satisfaction with environment friendly features, to be in line with ICT and other industrial products. The author justifies the development of green software by indicating some key factors of agile methods with relevant industrial practices.

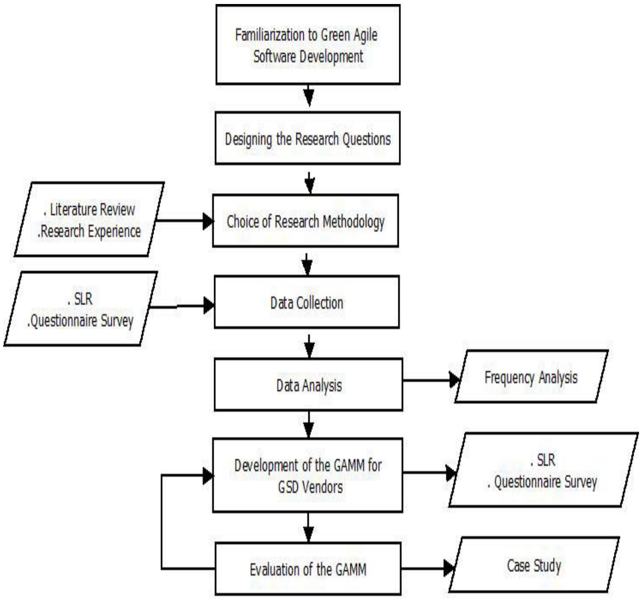
Familiarization to Green Agile Software Development
Designing the Research Questions
. Literature Review .Research Experience
Choice of Research Methodology
. SLR .Questionnaire Survey
Data Collection
Data Analysis
Frequency Analysis
Development of the GAMM for GSD Vendors
. SLR . Questionnaire Survey
Evaluation of the GAMM
Case Study

FIGURE 1. Stages of GAMM development.

TABLE 1. List of success factors identified through SLR.

S. No	Risk Factors	Frequency N=42	Percentage
1	Insufficient System Documentation	26	62
2	Limited Support for Real-Time Systems and Large Systems	19	45
3	Management Overhead	25	60
4	Lack of Customer’s Presence	26	62
5	Lack of Formal Communication	18	43
6	Limited Support for Reusability	07	17
7	Insufficient Knowledge of the Customer	15	36
8	Lack of Long-Term Planning	20	48

Galán*et al.* [43] argues the importance of sustainability in the context of organizational growth and emphasis that software development processes must be certified as environment friendly by implementing the agile practices. The authors propose an agile model for software development, using the old hardware platform, that could have long term use with considerable reduction of electronic waste.

III. RESEARCH QUESTIONS

RQ1. ‘‘What are the critical success factors, as identified in the literature, for adapting agile methods that can assist GSD vendors in the development of green and sustainable software’’?

RQ2: ‘‘What are the agile practices for the critical success factors, as identified in the literature, to be adapted by the

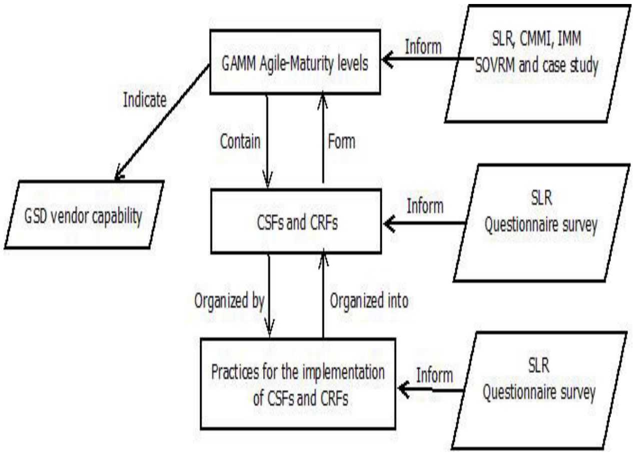

FIGURE 2. Structure of the GAMM.

GSD vendors in the development of green and sustainable software''?

RQ3. ''What are the critical risk factors, as identified in the literature, to be avoided by agile software developers in GSD for the development of green and sustainable software using agile methods''?

RQ4. ''How can a practical and robust Green Agile Maturity Model be developed''?

RQ5. ''Is the GAMM practically robust in terms of measuring organization's agile maturity for the development of green and sustainable software''?

IV. RESEARCH METHODOLOGY

The research methodology consists of the following phases:

Phase I (Conduct a Systematic Literature Review (SLR)): During this phase of the research study, we conducted SLR [44]–[50] to identify the critical success factors (CSFs) and critical risk factors (CRFs) in adapting agile methods for developing green software. Agile practices were also identified using SLR, to address the identified factors.

Phase II (Conduct a Questionnaire Survey): We conducted a questionnaire survey [51]–[54] in GSD industry. The participants included 106 agile experts from 25 different countries, for the results validation of Phase I and to probe other factors or practices.

Phase I and Phase II that answer the research questions RQ1-RQ3 is part of this research work, already carried out and available at [55]–[57], however, the main findings, i.e., success factors and risk factors are presented in Table 1 and Table 2 respectively, which provide an input for conduction of Phase III and Phase IV. This paper presents the implementation of Phase III and Phase IV only.

Phase III (Develop a Green-Agile Maturity Model (GAMM)): In this phase, we developed the GAMM, based on the inputs from SLR and questionnaire survey.

Phase IV (Conduct Case Studies): Case studies were conducted to evaluate the capability of the GAMM in software industry.

TABLE 2. List of risk factors identified through SLR.

S. No	Success Factors	Frequency *N=80*	Percentage
1	Accelerated Delivery	27	34
2	Continuous Integration	14	18
3	Continuous Validation	33	41
4	Efficient Utilization of time and Computing Resources	52	65
5	E-Waste Minimization	9	11
6	Flexibility Towards Change	29	36
7	Green and Sustainable Management of Product Life Cycle	34	43
8	Improved Quality	25	31
9	Iterative Development	31	39
10	Minimal Documentation	42	53
11	Minimal Reengineering	30	38
12	Optimization of Processes	5	6
13	Optimized Code	30	38
14	Polymorphic Design	38	48
15	Reduced Cost	17	21
16	Rich Communication and Collaboration	51	64

V. GREEN-AGILE MATURITY MODEL

A. DEVELOPMENT PROCESS OF THE GAMM

GAMM is based on the identified success factors and risk factors along with their identified practices through SLR and validated through questionnaire survey from 106 relevant international practitioners.

We have adapted the CMMI, IMM and SOVRM perspective [58]–[60] and developed the GAMM, as shown in Figure 2. The different stages of the GAMM development process are reflected in Figure 1. The fundamental proposed framework of the GAMM is available at [61] and well implemented and reported in this research work.

The success factors, risk factors and their relevant practices were gathered through well-known research methods SLR and industrial survey and these provided as inputs to the development of the GAMM. These factors were grouped together to form different levels as shown in Figure 2 and Figure 3.

GAMM was designed by assigning the success factors and risk factors to its relevant level. These GAMM's levels, described in Table 3, contain different CSFs and CRFs identified through the SLR and industrial survey. Under each factor different agile practices have been designed that guide how to assess each factor. Agile GSD vendors should address each factor to achieve a certain GAMM level.

In the first stage of GAMM development, criteria were developed, as mentioned below.

B. EVALUATION OF THE GAMM

A case study approach was used for evaluation of the GAMM. We aim to find five agile GSD vendor organizations

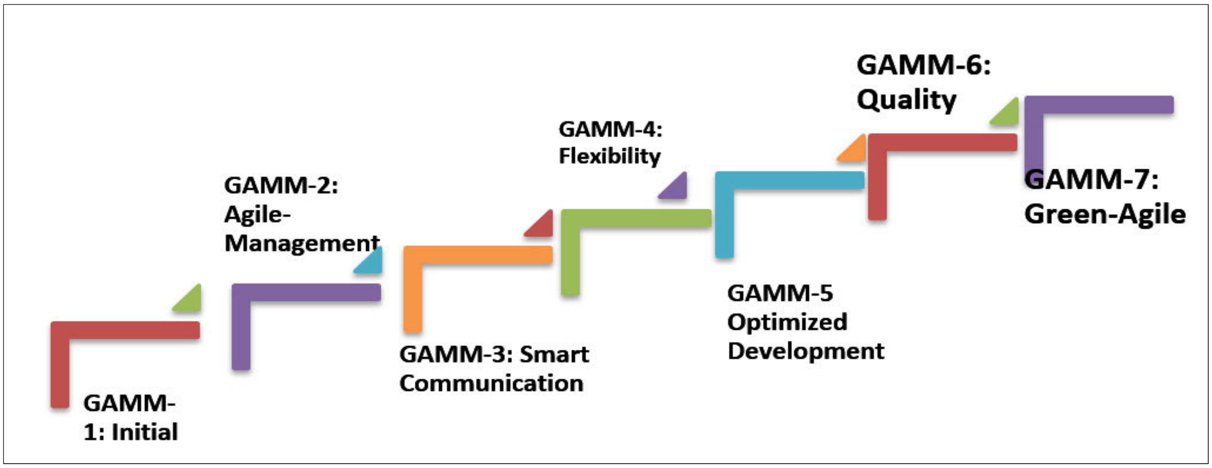

FIGURE 3. Green-agile maturity levels of the GAMM.

for evaluation of the GAMM. The same approach has been reported in [62], [63].The case study method was used to evaluate the GAMM, as it is considered a powerful evaluation tool and can provide useful real-world information [64]. The case study also produces valuable insights for problem solving, evaluation and strategy [65]. At the end of these case studies, feedback sessions were conducted with the participants to obtain feedback about the applicability of the GAMM.

C. STRUCTURE OF THE GAMM

The first phase of this study was to identify the critical success factors (CSFs) that influences the development process by adapting agile methods to deliver green software in GSD. We used mixed approach i.e., SLR followed by questionnaire survey to identify these factors. We have adapted CMMI [58], implementation maturity model (IMM) [59] and SOVRM [60] perspective and developed the GAMM. The structure of the GAMM is shown in Figure 2. The GAMM intends to assess the agile maturity of GSD vendor and will also provide complete guidance on how to overcome the risks for successful development of green software with agile methods. Figure 3 depicts the relation among GAMM levels, factors (CSFs) and various agile practices associated with each factor. It reveals how agile-maturity levels pinpoint GSD vendor's capability and how outcomes of the SLR and questionnaire survey provided input into GAMM levels. Most of the agile maturity levels comprise the different CSFs. However, level 1 contains no CSF.

There are 7 agile maturity levels in the GAMM, as presented in Figure 3. These levels consist of various CSFs and CRFs. To each CSF, different agile practices are assigned that will support how to implement each factor. The GAMM structure in Figure 2 shows that GSD vendors should implement each CSF to attain a certain agile maturity level. Structure of the GAMM is based on the following three components:

- Green-Agile levels component.
- CSFs and CRFs component.
- Assessment dimension.

The motivation for designing these components for the GAMM exude from CMMI, IMM and SOVRM perception. In CMMI, there are five different levels, which consist of various process areas (PAs). Each PA consists of different real-world practices. The categorization of CSFs in CMMI invigorated to design multi-level green-agile maturity model. These GAMM levels, as depicted in Table 3, contain several CSFs. For each factor, several agile practices are assigned that guide how to evaluate each factor.

D. COMPONENT LEVELS OF THE GAMM

CMMI has been developed, including five maturity levels i.e., level 1 to level 5. For the GAMM numerous modifications to this structure were found to be necessary to consider the specific GSD vendor's capabilities with agile practices to develop green software (as shown in Figure 3). This resulted in 7 levels for the GAMM, discussed as follows:

GAMM 1 (Initial): This level does not have a stable agile software development process.

GAMM 2 (Agile-Management): At this level of the GAMM, the focus is to motivate the agile development team to efficiently utilize the time and available computing resources for better software development. This level has only 1 CSF, as shown in Table 3.

GAMM 3 (Smart Communication): At this level of the GAMM, the focus is to enable the strong communication among the people by creating good communication channels through unified protocols within the development team and with customers. This level also emphasizes the production of reduced documentation, used for communication purposes. The introduction of these two agile properties in organization can induce the motivation of agile experts towards green-and-sustainable-software development. This level has 2 CSFs, as shown in Table 3.

GAMM 4 (Flexibility): At this level of the GAMM, the focus is to follow an iterative and incremental approach for software development. It also focuses on encouraging flexibility by accepting and accommodating the changes in

TABLE 3. Green-agile maturity levels of the GAMM.

GAMM levels	Objective	CSFs	CRFs
GAMM-7: Green-Agile	Objective of this level is to focus on green and sustainable management of product life cycle.	Green and sustainable management of product life cycle	Limited support for real time systems and large systems
GAMM-6: Quality	Objective of this level is to focus on the accelerated delivery of software to customer with improved quality.	Accelerated delivery	Nil
		Improved quality	
GAMM-5: Optimized Development	Objective of this level is to focus on development of reliable software with reduced code and simple design.	Polymorphic design	Nil
		Optimized code	
		Minimal reengineering	
		Continuous validation	
GAMM-4: Flexibility	Objective of this level is to emphasis on adopting the iterative approach and accepting the changes in software.	Iterative development	
		Flexibility towards change	Nil
GAMM-3: Smart Communication	Objective of this level is to focus on establishing a good communication among the team with reduced documentation.	Rich communication and collaborations	Insufficient system's documentation
		Minimal documentation	Lack of formal communication
			Lack of customer's presence
GAMM-2: Agile Management	Objective of this level is to focus on better utilization of time and available computing resources for efficient software development.	Efficient utilization of time and computing resources	Management overhead
			Lack of long-term planning
GAMM-1: Initial	Objective of this level is to investigate and develop stable agile software development process.	--------- Nil ------	

software generated from internal environment or from customer. This level has 2 CSF, as shown in Table 3.

GAMM 5 (Optimized Development): This level accentuates the development of the executable artefacts with optimized code, polymorphic design, minimal reengineering, and continuous validation, which are intrinsic factors of agile methods with effective input in developing green software. This level has 4 CSFs, as shown in Table 3.

GAMM 6 (Quality): This level of the GAMM emphasis on the accelerated delivery of software to customers with improved quality. This level has 2 CSFs, as shown in Table 3.

GAMM 7 (Green-Agile): At this level of the GAMM, the focus is on the establishment of sustainable management of product life cycle. This level has 1 CSFs, as shown in Table 3.

The designed levels for the GAMM proved its purpose and efficiency required for classifying the green-agile maturity of a particular GSD vendor. The idea of categorising the levels for the GAMM is adapted from CMMI levels structuring, factors (SFs) and practices identified through SLR and from agile experts. Some researchers also adapted the same procedure for model development [60], [63].

This model measures the green-agile maturity of a GSD vendor with respect to green and sustainable software development.

1) THE SUCCESS FACTOR COMPONENT

The CMMI structure reflects different process areas (PAs), categorized into five levels of maturity. Maturity of software processes could be examined through the defined PAs. In this research study, we have viewed the green-agile maturity in terms of the implementation of the identified factors not the PAs. The critical success factors of green-agile maturity are complementary to CMMI's PAs. The notion of using CSFs and CBs as measuring variables has been implemented in other research as well [59], in which the CSFs and CBs are based for process improvement maturity model. Similarly, Khan [60] has adopted the same technique in the development of outsourcing vendor's readiness model. Similar approach has been followed by [63].

Keeping in view these approaches for models development, the identified factors and practices were categorized to design the different levels for the GAMM, like the categorization of PAs into four broad categories in CMMI i.e. project management, process management, engineering and support [66]. This classification of the CSFs and CRFs steered us to design seven agile maturity levels, i.e., 'Initial', 'Agile management', 'Smart communication', 'Flexibility', 'Optimized development', 'Quality' and 'Green-agile'. These levels with concerned factors are depicted in Table 3. The levels are devised according to cohesion among the identified CSFs and CRFs. This categorization procedure has been assessed through five case studies.

2) ASSESSMENT OF THE GAMM

In this component of the GAMM, all the CSFs and CRFs are measured to evaluate the extent to which these factors have been implemented in practice by the GSD vendors. To measure the implementation of a particular agile factor Motorola [67] assessment tool has been used to assess the

TABLE 4. Evaluation dimensions of Motorola assessment instrument.

Approach	Criteria for this dimension is the management support and organization commitment for the practice as well as the organization's capacity to implement that practice
Deployment	Criteria for this dimension is the breadth and flexibility
	for the implementation of a practice across all project sphere
Results	Criteria here are the breadth and firmness of positive outcomes with the passage of time across the project areas

TABLE 5. Steps for evaluation of factors using Motorola instrument.

S. No	Evaluation steps
1	For each calculate the 3-dimensional score of the assessment instrument
2	Add 3-dimensional scores for each practice, then divide by 3 and round to nearest whole number
3	Repeat this process for each practice. Add the scores of each practice and calculate average for each factor

current status of processes. This method is simple to understand and implement with a limited set of activities. This assessment method is tried and tested and allows software engineers and managers to assess the organization's present status relative to CMM and pinpoints the weak areas that require more focus for improvement.

It emulates the CMM concept of measuring the levels of process maturity through a checklist. This checklist is a methodological way to measure the degree to which a recommended process is present in an organisation and how useful it is for that organisation. It is a straightforward approach of gauging the process maturity in the organizations [68].

This instrument evaluates each key process area (KPA) as a score from 1 to 10, which is then calculated as an average-score for every KPA. For any KPA, with an average score below 7 is treated as a weakness. The same assessment instrument has been followed for the evaluation of "Software outsourcing vendors' readiness model" (SOVRM) [60].

The different criteria/key points considered for evaluation in Motorola assessment instrument are shown in Table 4.

Against each evaluation criteria, a score (1-10) is allocated, based on the guidelines. For each CSF and CRF, we have identified an index of practices.

The steps, as shown in Table 5 are conducted to assess the GAMM through Motorola instrument, an example is presented in Table 6).

Agile practices are associated with each factor, while the factors are associated with each level of the GAMM. It was not possible to include all the practices for addressing each factor in the manuscript, however we have mentioned the practices for only one factor, Efficient utilization of time and computing resources in Table 6. For more such practices for each factor of a particular level of the GAMM, detailed practices are published in [56].

To associate the evaluation points to the GAMM levels: score $\geq$ 7 means completely addressed the factors while, score below than 7 means a factor is partially addressed and reflects a weakness. To reach a particular GAMM level, it is mandatory all factors of that level have score 7 or above. For instance, to be ranked in GAMM Level 4, all the factors lying in this level must have score 7 or above.

VI. EVALUATION OF THE GAMM

To evaluate the GAMM, five case studies were conducted at GSD organizations. Each case study was evaluated by a panel of 10 experts which comprised representatives from industry and research

A. EVALUATION CRITERI

The primary motivation for designing the evaluation criteria is adopted from the IMM [59], SOVRM [60] and SOP [63]. Further motivation has been gained from the literature [25], [69], [70] and by considering the Technology Acceptance Model [71]. The below mentioned criteria is considered sufficient, as it will be adopted to measure the efficacy and quality of the GAMM and will point out the strengths and weaknesses of the GSD vendors regarding the agile maturity that need in-depth focus of the management.

The evaluation criteria for the GAMM are described as follows:

1) EASE OF USE

It is quite challenging for the GSD vendors to adopt the complex available models and procedures to meet the frequent changes in requirements from the customers as well as to meet the current needs for green and sustainable software development. The GAMM is designed in a way to be adopted with comfort for developing green software with agile methods by the GSD vendors.

2) EFFICACY OF THE GAMM

The aim is to analyse the performance and strengths of various modules of the GAMM and to further validate the dissemination of CSFs and CRFs across its different maturity levels.

Evaluation of the GAMM is quite essential to explore the different areas where the final product has deficits. The assessment helps in better decisions for organizations' future. The lessons learned from the assessment mechanism are considered for improving the GAMM in future.

B. EVALUATION THROUGH CASE STUDIES

There are good reasons for adopting the case study because this technique is considered more significant for model assessment and can provide appropriate information, as case study are grounded on lived reality [72]. The case study also facilitates the rich theoretical development and produces valuable understanding for complex structure of what is going to be studied.

The GAMM poses the features to assess the GSD vendors and is more appropriate to evaluate the green-agile maturity, so in this regard the case study is better to evaluate the

TABLE 6. Factor evaluation example.

CSF1: Efficient Utilization of Time and Computing Resources					
		Key Activity Evaluation Dimension			
S. No	Practices	Approach Score-range: 0,2,4,6,8,10	Deployment score-range: 0,2,4,6,8,10	Results score-range: 0,2,4,6,8,10	Average Score (Average-of three-dimensional values)
1	Avoid defect backlogs from the early days for each software development	10 (e.g., Management provides enthusiastic leadership and commitment)	8 (e.g., Deployed across the whole project)	6 (e.g., Consistently provides positive results over-time in all organization)	(10+8+6)/3=8
2	Use the existing tools for software development such as Jira, Axosoft on time Scrum, LeanKit etc.	8	6	4	6
3	Always code the software in pairs (pair programming)	10	8	6	8
4	Plan a schedule for the increments release to customer	10	8	6	8
5	Follow short development cycles/iterations	10	6	4	7
6	Develop re-usable prototype instead of throw –away prototype	10	8	6	8
7	Focus on rapid software development by spending less time in analysis and design phases	10	8	6	8
8	Track the velocity of development to deliver software in time	10	8	8	8
Sum of average score:					53
Overall score					7

GAMM through software industry practitioners. Case studies are imperative for the evaluation of the GAMM because they:

- Revealed that the GAMM is apposite to fit in the agile GSD vendor organizations.
- Pin-pointed the weak areas of the GAMM that needs to be more focused for improvements.
- Showed the applicability and usability of the GAMM in agile development industry.

The case study organizations were provided a guidance document for assessment process of the GAMM. The case study document presented various sections i.e., introduction to GAMM with detailed overview about each level, their role as a participant of the case study, overview of case study assessment process with an example through Motorola assessment tool. Beside these details the participants were provided a list of agile practices, identified through SLR, and validated through questionnaire survey, against each factor of different levels of the GAMM, as how these practices are applied (approach, deployment and resulted impact) to calculate the average score of each factor implementation.

1) CASE STUDY CONDUCTED AT COMPANY A

Company-A (CMMI-level 3) is a software development and IT consultancy company having more than 600 professionals in different countries. Businesses and individuals find Company-A an indispensable partner in successfully growing IT and IT-related businesses. It provides high quality ICT solutions to international market, implementing state-of-the-art exclusive software products, custom software development services as well as IT consultancy.

Company-A's technical and experienced professionals can provide effective information technology solutions to help companies achieve a competitive advantage with emphasis on:

- Desktop applications
- Project management trainings
- Mobile applications
- eHealth and mHealth products
- Games development
- ERP solutions
- Web development

2) GAMM ASSESSMENT RESULTS AT COMPANY A

To measure the agile maturity of the GSD vendor, we have considered the assessment component of the GAMM, based on the Motorola assessment instrument [67]. According to Motorola tool, if any factor scores 7 or high, will be considered as strong factor, as the agile practices are well implemented for adopting that factor, while a factor with score less than 7 is considered as weak factor, as the relevant agile practices are weakly implemented for adopting that factor.

A panel of 5 senior software engineers of this Company practiced the GAMM and has easily measured the green-agile maturity of his/her company. Table 7 depicts summarized evaluation of Company A.

Major points of this assessment are as follow:

The results, as shown in Table 7, signify that Company-A is positioned at Level-2 'Agile Management' of the GAMM because only one factor of Level-3 'insufficient system's documentation' has a score less than 7. To attain any level of the GAMM, it is essential that all the related factors of that level must have an average score 7 or above. Table 7 indicates that to achieve Level-3 'Smart Communication' of the GAMM, company-A requires to progress only one factor i.e.,

TABLE 7. GAMM assessment results at company A.

GAMM levels	CSFs and CRFs	Score	Status
GAMM-7: *Green-Agile*	Green and sustainable management of product life cycle	6	Weak
	Limited support for real time systems and large systems	5	Weak
GAMM-6: Quality	Accelerated delivery	6	Weak
	Improved quality	5	Weak
GAMM-5: *Optimized Development*	Polymorphic design	7	Strong
	Optimized code	7	Strong
	Minimal reengineering	7	Strong
	Continuous validation	6	Weak
GAMM-4: Flexibility	Iterative development	7	Strong
	Flexibility towards change	5	Weak
GAMM-3: *Smart Communication*	Rich communication and collaboration	8	Strong
	Insufficient system's documentation	5	Weak
	Minimal documentation	8	Strong
	Lack of formal communication	8	Strong
	Lack of customer's presence	8	Strong
GAMM-2: *Agile Management*	Efficient utilization of time and computing resources	8	Strong
	Management overhead	8	Strong
	Lack of long-term planning	7	Strong
GAMM-1: Initial	--------- Nil ------		

'insufficient system's documentation'. Similarly, to be ranked in Level-4 of the GAMM i.e., 'Flexibility', the Company-A requires to progress one factor, i.e., 'Flexibility towards change'.

Company-A has implemented most of the factors, as shown in Table 7 to develop green and sustainable software. More than half of the factors have scored >= 7. The remaining factors can be improved with little efforts to achieve higher agile-maturity levels of the GAMM.

3) CASE STUDY CONDUCTED AT COMPANY B

Company-B (CMMI level-3) is an IT solution provider with a wide range of partners across the globe. Company-B has a good number of agile- PMP certified team members. It is specialized in delivering key IT data security, storage, and systems management solutions to help its customers in different sectors. It is one of the few companies at Pakistan that offer a wide range of software product and IT solution to cover various IT needs across the industries.

Company-B has a wide range of services and solutions and the up-to-date technical expertise. It has sound ability to understand their clients' requirements and technological resources to create a system that supports it. Company B's technical and experienced professional can provide effective information technology solutions to help companies achieve a competitive advantage with emphasis on:

- Storage and data protection solutions
- Data security
- VAS solution
- Virtualization
- Education and training
- IOS development
- Professional services
- Managed services
- IT reseller
- System integration
- E-commerce

4) GAMM ASSESSMENT RESULTS AT COMPANY B

To evaluate the agile-maturity of the GSD vendor, we have considered the assessment component of the GAMM, based on the Motorola assessment instrument [67]. According to

TABLE 8. GAMM assessment results at company B.

GAMM levels	CSFs and CRFs	Score	Status
GAMM-7: Green-Agile	Green and sustainable management of product life cycle	5	Weak
	Limited support for real time systems and large systems	5	Weak
GAMM-6: Quality	Accelerated delivery	5	Weak
	Improved quality	4	Weak
GAMM-5: Optimized Development	Polymorphic design	7	Strong
	Optimized code	7	Strong
	Minimal reengineering	7	Strong
	Continuous validation	8	Strong
GAMM-4: Flexibility	Iterative development	8	Strong
	Flexibility towards change	7	Strong
GAMM-3: Smart Communication	Rich communication and collaborations	8	Strong
	Insufficient system's documentation	8	Strong
	Minimal documentation	8	Strong
	Lack of formal communication	7	Strong
	Lack of customer's presence	8	Strong
GAMM-2: Agile Management	Efficient utilization of time and computing resources	8	Strong
	Management overhead	8	Strong
	Lack of long-term planning	8	Strong
GAMM-1: Initial	--------- Nil ------		

this module, if any factor scores 7 or higher, will be considered well implemented at that company, while a factor with score less than 7 is considered as weakly implemented factor.

A panel of 7 experts used the GAMM and measured the green-agile maturity of the Company-B. Table 8 depicts summarized evaluation of Company B.

The results, as shown in Table 8 signify that Company-B is positioned at Level-5 'Optimized Development' of the GAMM because all the factors below this level have been successfully implemented. Table 8 indicates that to achieve the highest Level-7 'Green-agile' of the GAMM, the Company-B requires to progress few factors, marked as weak.

The results, as presented in Table 8, manifest that Company-B is an established IT company with matured agile processes to meet the requirements for the development of green and sustainable software development.

5) CASE STUDY CONDUCTED AT COMPANY-C

Company-C is an information technology company (CMMI level-2) that provides custom software development, IT consultancy and application outsourcing services. It has a team of senior experts using their innovative plat forming approach to provide services to clients. Company-C ensures high quality cost effective and adequate solutions that enable the clients to amplify the business operations and productivity. Company C is an innovator in providing the high-quality web-based solutions to diverse companies, through a team of genius programmers, marketing executives and designers.

Company C provides the following services and products:

- Enterprise solution
- Corporate finance
- ERP consulting
- CRM
- Dashboard reports development
- Web development
- Mobile applications
- Digital healthcare

6) GAMM ASSESSMENT RESULTS AT COMPANY C

To evaluate the agile-maturity of the GSD vendor, we have considered the assessment component of the GAMM, based on the Motorola assessment instrument [67]. According to this module, if any factor scores 7 or higher, will be considered well implemented at that company, while a factor with score less than 7 is considered as weakly implemented factor.

A panel of 7 experts of high-profile associates of the agile development society, Pakistan has used the GAMM and measured the green-agile maturity of the Company-C. Table 9 portrays Company C's evaluation scores.

TABLE 9. GAMM assessment results at company C.

GAMM levels	CSFs and CRFs	Score	Status
GAMM-7: Green-Agile	Green and sustainable management of product life cycle	5	Weak
	Limited support for real time systems and large systems	5	Weak
GAMM-6: Quality	Accelerated delivery	7	Strong
	Improved quality	5	Weak
GAMM-5: Optimized Development	Polymorphic design	7	Strong
	Optimized code	8	Strong
	Minimal reengineering	8	Strong
	Continuous validation	6	Weak
GAMM-4: Flexibility	Iterative development	8	Strong
	Flexibility towards change	5	Weak
GAMM-3: Smart Communication	Rich communication and collaborations	7	Strong
	Insufficient system's documentation	5	Weak
	Minimal documentation	5	Weak
	Lack of formal communication	4	Weak
	Lack of customer's presence	6	Weak
GAMM-2: Agile Management	Efficient utilization of time and computing resources	8	Strong
	Management overhead	8	Strong
	Lack of long-term planning	9	Strong
GAMM-1: Initial	--------- Nil ------		

The results presented in Table 9, indicate that Company-C is positioned at Level-2 'Agile management' of the GAMM as 4 factors of Level-3 are completely addressed by the Company C. Table 9 also indicates that for Company C to achieve Level-3 of the GAMM, it must progress to address the factors, marked as weak. The results of the Company-C, as shown in Table 9, signifies that Company-C can achieve higher levels of the GAMM, if the management focus more on addressing the factors at GAMM levels 6 and 7.

7) CASE STUDY CONDUCTED AT COMPANY-D

Company-D (CMMI-level 2) is an international software and IT service company with its partners in more than 15 different countries around the globe. Company-D is equipped with strongly committed 24/7 real people Gold certified support, serving the clients with the latest engineered products.

Company-D provides the following services and products to its clients.

- Web development
- IOS development
- Graphics designing
- Web hosting
- Desktop applications
- Mobile applications
- BPO solutions

8) GAMM ASSESSMENT RESULTS AT COMPANY-D

To evaluate the agile-maturity of the GSD vendor, we have considered the assessment component of the GAMM, based on the Motorola assessment instrument [67]. According to this module, if any factor scores 7 or higher, will be considered well implemented at that company, while a factor with score less than 7 is considered as weakly implemented factor.

A panel of 6 senior software engineers of this Company practiced the GAMM and has easily measured the green-agile maturity of his/her company. Table 10 depicts summarized scores of Company-D.

The results, shown in Table 10, portray that Company-D is positioned at Level-5 'Optimized development' of the GAMM because one factor 'Improved quality' of Level-6 'Quality' is not fully implemented at Company-D. Table 10 reveals that to achieve Level-7 'Green-Agile' of the GAMM, the Company-D requires to progress few, marked as weak. Table 10 illustrates that Company-D has implemented most of the factors which indicates its matured agile processes through experience agile team and hence can better develop green and sustainable software.

9) CASE STUDY CONDUCTED AT COMPANY-E

Company-E is an IT company that provides a broad range of technological services to all sorts of business organizations through its qualified and experienced team members.

TABLE 10. GAMM assessment results at company-D.

GAMM levels	CSFs and CRFs	Score	Status
GAMM-7: Green-Agile	Green and sustainable management of product life cycle	6	Weak
	Limited support for real time systems and large systems	5	Weak
GAMM-6: Quality	Accelerated delivery	8	Strong
	Improved quality	6	Weak
GAMM-5: Optimized Development	Polymorphic design	8	Strong
	Optimized code	7	Strong
	Minimal reengineering	8	Strong
	Continuous validation	7	Strong
GAMM-4: Flexibility	Iterative development	7	Strong
	Flexibility towards change	7	Strong
GAMM-3: Smart Communication	Rich communication and collaborations	8	Strong
	Insufficient system's documentation	7	Strong
	Minimal documentation	8	Strong
	Lack of formal communication	9	Strong
	Lack of customer's presence	9	Strong
GAMM-2: Agile Management	Efficient utilization of time and computing resources	8	Strong
	Management overhead	8	Strong
	Lack of long-term planning	8	Strong
GAMM-1: Initial	--------- Nil ------		

Company-E is at CMMI- level 3. It has partners in 10 different countries. Company-E is a rich profile technology company that add factual values to their client's business through accelerated and innovative delivery of software products and IT services with exclusive technology experts.

Company-E emphasis on provision of the following services to its clients.

- Biometric verification systems
- Business management and transformations
- Banking financial services
- Image processing solutions
- Mobile applications
- Desktop applications
- ERP consulting
- Social media marketing

10) GAMM ASSESSMENT RESULTS AT COMPANY-E

To evaluate the agile-maturity of the GSD vendor, we have considered the assessment component of the GAMM, based on the Motorola assessment instrument [67]. According to this module, if any factor scores 7 or higher, will be considered well implemented at that company, while a factor with score less than 7 is considered as weakly implemented factor.

A panel of 6 experts including country manager of this Company practiced the GAMM and has easily measured the green-agile maturity of his/her company. Table 11 illustrates summarized scores of Company-E. The results described in Table 11, signifies that Company-E is positioned at level-4 'Flexibility' of the GAMM because two factors of level-5 'optimized development' are partially addressed by the company. As shown in Table 11, for Company-E to achieve level-5 of the GAMM, requires improving two factors i.e., 'optimized code' and 'continuous validation'. To attain level-6 'Quality' and level-7 'Green-agile' the Company-E requires to advance some factors, marked as weak.

From the factor's evaluation at Company-E, it is apparent that Company-E places at quite good level of the GAMM. The reason being that it has been established 6 years ago but due to experienced agile development team, it has attained level-4 of the GAMM and has the potential to achieve higher levels if the management of the Company focus more on addressing their weak factors at level 5-7.

VII. FEEDBACK SUMMARY

We got a feedback from the case study participants to evaluate the basic criteria of the GAMM i.e., ease of use, users' satisfaction and about its structure. The feedback of the five case study organizations, summarized and portrayed in Tables 7-11, demonstrate entire satisfaction on the designed metrics. The case study participants also agreed

TABLE 11. GAMM assessment results at company E.

GAMM levels	CSFs and CRFs	Score	Status
GAMM-7: Green-Agile	Green and sustainable management of product life cycle	4	Weak
	Limited support for real time systems and large systems	5	Weak
GAMM-6: Quality	Accelerated delivery	7	Strong
	Improved quality	4	Weak
GAMM-5: Optimized Development	Polymorphic design	8	Strong
	Optimized code	6	Weak
	Minimal reengineering	7	Strong
	Continuous validation	6	Weak
GAMM-4: Flexibility	Iterative development	8	Strong
	Flexibility towards change	8	Strong
GAMM-3: Smart Communication	Rich communication and collaborations	8	Strong
	Insufficient system's documentation	8	Strong
	Minimal documentation	8	Strong
	Lack of formal communication	9	Strong
	Lack of customer's presence	7	Strong
GAMM-2: Agile Management	Efficient utilization of time and computing resources	8	Strong
	Management overhead	7	Strong
	Lack of long-term planning	8	Strong
GAMM-1: Initial	-------- Nil ------		

TABLE 12. Feedback results (Ease of use) of five different companies.

Ease of Use	Organizations' Perception (N=5)				
	Strongly Agree	Agree	Disagree	Strongly Disagree	Not Sure
GAMM representation is very clear	4	1	Nil		
A little knowledge of agile software development is required to learn how to use the GAMM	3	1			
It is easy to understand the practices designed for each critical success factors and critical risk factors	3	2			
It is easy to understand the assessment method	4	1			
It is easy to use the GAMM to assess organizations agile maturity for green and sustainable software development	3	1			
It is easy to understand distribution of critical success factors and critical risk factors among different agile maturity levels.	3	2			
Some training needs to be provided for the use of GAMM.	2	2	1	0	0

with the results about their green-agile maturity level generated through the GAMM. The participants at the five companies have shown complete satisfaction in the GAMM with respect to ease of use, end user satisfaction, structure, and its practices for CSFs and CRFs. Different questions were used to enquire about ease of use. It was revealed that the participants' general impression to use the GAMM was very positive. The participants agreed with the results of assessment. Overall, the participants fully agreed with the ease of use of different components within the GAMM. Different questions were enquired about user satisfaction. It was revealed that the participants' general impression of user satisfaction was very positive, e.g., the participants' positive responses regarding the generality of the GAMM are 100%. The participants' responses regarding ease of use, end user satisfaction, and structure of the GAMM are shown in Table 12, Table 13 and Table 14 respectively.

Table 15 reflects the responses for the open-ended question. The results are quite encouraging and demonstrates that the GAMM could be adapted by the GSD vendors,

TABLE 13. Feedback results (End-user satisfaction) of five different companies.

End-User Satisfaction	Organizations' Perception (N=5)				
	Strongly Agree	Agree	Disagree	Strongly Disagree	Not Sure
GAMM is general and can be applied to most companies	3	2	Nil		
Each individual practice is easy to understand and unambiguous	4	1			
Using the GAMM would identify strong and weak areas in the company regarding agile maturity for green and sustainable software development	3	2			
Using the GAMM would improve our agile maturity for green and sustainable software development	4	1			
If the GAMM were available for my job, I predict that I would use it on regular basis in the future.	2	2	0	0	1
I am satisfied and agreed with the agile maturity issues identified by the GAMM	4	1	Nil		
It is important to implement GAMM in the form of an automated software tool in order to facilitate software practitioners in assessing organization's agile maturity	3	2	Nil		
The GAMM is self-contained	3	2			
The assessment method is useful	4	1			

TABLE 14. Feedback results (Structure of the GAMM) of five different companies.

Structure of the GAMM	Strongly Agree	Agree	Disagree	Strongly Disagree	Not Sure
All the components of the GAMM are self-explanatory and require no further explanation to be used effectively	3	2	Nil		
The components of the GAMM are practical and are applicable in GSD industry	3	2			
The GAMM can be used effectively to identify agile maturity issues with a goal to develop green and sustainable software development.	4	1			
The distribution of critical success factors and critical risk factors among different agile maturity levels is useful	4	1			
The 7 agile maturity levels of the GAMM are useful	3	2			

as it provides a practical tool for assessing the green-agile maturity.

VIII. LIMITATIONS OF RESEARCH DESIGN

In this research, SLR is used for the identification of the CSFs and CRFs in green and sustainable software development, using agile methods. We have extracted data from a sample of 122 publications. With the increasing number of papers in agile software development, the SLR process may have missed out some relevant papers. However, like other researchers of SLR this is not a systematic omission [73].

The questionnaire survey explored the perceptions and experiences of agile experts regarding the factors (CSFs and CRFs) and their concerned agile practices for the development of green and sustainable software. These perceptions and experiences have not been verified directly. This may mean that what agile experts say about the critical factors may not necessarily be the critical factors for GSD vendors' agile maturity. Furthermore, agile experts' perceptions may not be accurate.

To validate the GAMM and measure its applicability in industry, we conducted five case studies at different GSD organizations. To deduce conclusions from this small and idiosyncratic sample of the GSD vendors may lead to generalizations, which may not be applicable to all GSD vendors. Additional exploratory studies of the same kind are needed, so that the GSD community is enlightened with how agile experts adopt the different agile practices for the development of green and sustainable software. Some limitations of the case studies are as follow:

- As the nature of data is qualitative, some portions of the case study cannot be described in numerical form, such as the GSD vendors' feedback was analysed qualitatively.
- It is quite hard and time consuming to gather the case study data and even more difficult to analyse.
- The GAMM maturity level, attained by a specific GSD vendor, is not asserted by researcher as the actual one for that GSD vendor, because the facts provided by the case study participants were not autonomously confirmed from other sources. By default, we rely on the case study

TABLE 15. Feedback results (open ended questions).

Open ended questions	Organizations' Perception (N=5)					Average Response
	Company A	Company B	Company C	Company D	Company E	
Are there any modifications or improvements to the GAMM that you may suggest?	I suggest that some more factors of agile methods should be explored, if possible, to increase the efficacy of software development.	It is good at this initial stage.	GAMM is unique in its feature. It is more than enough at this stage.	It is a unique model. It has more than enough levels.	I don't think so. It is enough.	Very positive
Are there any components that you may suggest adding to the GAMM in the future, please also give the reasons?	No	No	No	No	Enough components. It has covered all the aspects of agile software development.	Very positive
Please provide any comments relating to the assessment method.	The assessment method is simple and easy to follow. It covers all the dimensions to measure the agile-maturity of software organization.	The assessment method is quite simple	The assessment method is very simple to understand.	The assessment method is quite useful, and one can easily follow it with short training.	It is ok.	Very positive
Please provide any comments relating to the distribution of practices across various critical success factors and critical risk factors.	All the identified factors practices are well distributed.	A good number of practices have been explored and ell distributed across the different levels	All the agile practices are well distributed.	All the factors are well sorted among the different levels.	. Well distributed.	Very positive
Please provide any comments relating to the usability of the GAMM with respect to time it takes users to measure agile maturity of GSD vendors.	It is efficient with respect to time. Al the software development organizations must know about the GAMM for implementation	GAMM is very much clear	It seems to be very helpful if implemented in software company. One can easily assess their organization's agile maturity for green development.	It takes short time to measure the agile maturity, as all the factors and practices are quite clear.	It is very simple and needs feasible time.	Very positive

participants to be equitable in reporting the data for case study.

- Green-agile maturity is evaluated by a single assessment technique (Motorola instrument).
- Virtuous relationship was created with the case study participants of GSD organizations. In response, the case study participants may have provided optimistic comments about the GAMM.

IX. KEY CONTRIBUTION TO KNOWLEDGE

This research study makes a practical contribution to help GSD vendors for quantifying their maturity in developing green software with agile methods by devising a Green Agile-Maturity Model (GAMM) for GSD vendors. This model will contribute to the knowledge in Agile GSD. The GAMM will bring together and progress the work, carried out for frameworks and models for agile software development. The evaluation through five case studies confirmed that the GAMM is useful in real industry regarding the evaluation of agile maturity of GSD vendors for green and sustainable software development.

This research study also makes a methodological contribution by adopting SLR as a research strategy to spot out the CSFs and CRFs in developing green software by practicing agile methods in GSD. This is because no systematic literature review has been carried out to date on agile practices in general and the identification of factors that have a significant impact on GSD vendor organizations in the context of developing green software with agile methods. The lack of such maturity model to analyze the capability of vendor organization in developing green software with agile methods evince the need of and confirms the novelty of our research work. An examination of the relevant literature through SLR, together with a questionnaire survey, led us to design and develop the GAMM. The development process of the GAMM has full transparency and we have shown explicitly how the GAMM is designed, developed, and evaluated. This constructs a methodological contribution by providing methodological process which can be re-used for the development of other models by other researchers. The contribution to improving agile maturity will provide other researchers with a firm basis on which to develop different agile practices that are based on an understanding of how and where they fit into the green and sustainable software development activities. New agile practices could then be developed targeting the development of green and sustainable software.

X. CONCLUSION AND DIRECTIONS FOR FUTURE WORK

The novel model presented in this paper contributes some knowledge to the green-agile in GSD domain. The GAMM is developed to measure the green-agile maturity of GSD vendor with respect to green and sustainable software development. The GAMM also dig out the problems in practicing and implementing agile practices through distinguished designing strategy, rarely adapted for maturity models, and reported in literature.

Our main contribution to software engineering knowledge is the development of the GAMM, which will assess the agile maturity of GSD vendors regarding developing of green and sustainable software with agile methods.

In future, we plan to work out for more strengthening the GAMM by incorporating more levels with increased number of CSFs, CRFs and associated practices that would stress the agile team to evaluate the green-agile maturity of an individual member at the GSD vendors. It will introduce a new version of the GAMM i.e., Green-Agile Maturity Model-Team Evaluation (GAMM-TE). The GAMM-TE will help to identify the problems at agile team level and will assess the green-agile mind-set to better evaluate the overall GSD vendor.

According to suggestions of the case study participants, in future the GAMM will be provided as a wed-based application for transparent and better evaluation of the GSD vendors, to assess their agile maturity regarding development of green and sustainable software with agile methods. The GAMM software tool will be a web-based application enriched with advanced features to reflect the model requirements and will generate required reports, such as the green-agile level of a particular GSD vendor, based on the provided inputs as key scores for addressing the success factors and risk factors through its implemented practices. The GAMM application will provide:

Detailed analysis of the capabilities of the GSD vendor.

Recording the results of assessment of each CSF and CRF, regarding the agile maturity.

Identifying the weak and strong factors of a particular GSD vendor.

Identifying the agile-maturity of GSD vendors for green-and- sustainable-software development

Generating different assessment reports regarding the green agile maturity.

Software tool automates the functions and enhances the processes' visibility, pinpoints the organization's weaknesses, and helps to improve the productivity. The GAMM software will examine the nature and performance of various agile practices for addressing the different factors. The case study experts anticipated that the software tool, if developed, would faster the whole assessment process and functions of the GAMM.

ACKNOWLEDGMENT

The findings achieved herein are solely the responsibility of the authors.

CONFLICT OF INTEREST

The authors declared that there are no potential conflicts of interests regarding this article.

REFERENCES

[1] R. M. Fontana, V. Meyer, S. Reinehr, and A. Malucelli, "Progressive outcomes: A framework for maturing in agile software development," *J. Syst. Softw.*, vol. 102, pp. 88–108, Apr. 2015.

[2] M. Lindvall, V. Basili, B. Boehm, P. Costa, K. Dangle, F. Shull, R. Tesoriero, L. Williams, and M. Zelkowitz, "Empirical findings in agile methods," in *Extreme Programming and Agile Methods—XP/Agile Universe 2002*. Berlin, Germany: Springer-Verlag, 2002, pp. 197–207.

[3] S. Misra, V. Kumar, U. Kumar, K. Fantazy, and M. Akhter, "Agile software development practices: Evolution, principles, and criticisms," *Int. J. Qual. Rel. Manage.*, vol. 29, no. 9, pp. 972–980, Oct. 2012.

[4] M. Fowler and J. Highsmith, "The agile manifesto," *Softw. Develop.*, vol. 9, no. 8, pp. 28–35, 2001.

[5] J. Sheffield and J. Lemétayer, "Factors associated with the software development agility of successful projects," *Int. J. Project Manage.*, vol. 31, no. 3, pp. 459–472, Apr. 2013.

[6] L. Vijayasarathy and D. Turk, "Drivers of agile software development use: Dialectic interplay between benefits and hindrances," *Inf. Softw. Technol.*, vol. 54, no. 2, pp. 137–148, Feb. 2012.

[7] S. Agarwal, A. Ghosh, S. Bhattacharyya, and A. Nath, "Green solutions: A pilot study on green technology and green computing," *Int. J.*, vol. 5, no. 10, pp. 680–686, 2015.

[8] S. Agarwal, A. Nath, and D. Chowdhury, "Sustainable approaches and good practices in green software engineering," *Int. J. Res. Rev. Comput. Sci.*, vol. 3, no. 1, pp. 1425–1428, 2012.

[9] M. Dick, S. Naumann, and N. Kuhn, "A model and selected instances of green and sustainable software," in *What Kind of Information Society? Governance, Virtuality, Surveillance, Sustainability, Resilience*. Berlin, Germany: Springer, 2010, pp. 248–259.

[10] M. Dick and S. Naumann, "Enhancing software engineering processes towards sustainable software product design," in *Proc. EnviroInfo*, 2010, pp. 706–715.

[11] S. S. Shenoy and R. Eeratta, "Green software development model: An approach towards sustainable software development," in *Proc. Annu. IEEE India Conf.*, Dec. 2011, pp. 1–6.

[12] M. Thiry, L. Frez, and A. Zoucas, "GreenRM: Reference model for sustainable software development," in *Proc. SEKE*, 2014, pp. 39–42.

[13] F. Albertao, "Sustainable software development," in *Harnessing Green IT*. Hoboken, NJ, USA: Wiley, 2012, pp. 63–83.

[14] A. Raturi, B. Tomlinson, and D. Richardson, "Green software engineering environments," in *Green in Software Engineering*. Cham, Switzerland: Springer, 2015, pp. 31–59.

[15] S. Naumann, M. Dick, E. Kern, and T. Johann, "The greensoft model: A reference model for green and sustainable software and its engineering," *Sustain. Comput., Inform. Syst.*, vol. 1, no. 4, pp. 294–304, 2011.

[16] A. Govindasamy and K. S. Joseph, "Optimization of operating systems towards green computing," *Int. J. Combinat. Optim. Problems Inform.*, vol. 2, no. 3, pp. 39–45, 2011.

[17] C.-T.-D. Lo and K. Qian, "Green computing methodology for next generation computing scientists," in *Proc. IEEE 34th Annu. Comput. Softw. Appl. Conf.*, Jul. 2010, pp. 250–251.

[18] K. Erdelyi, "Special factors of development of green software supporting eco sustainability," in *Proc. IEEE 11th Int. Symp. Intell. Syst. Informat. (SISY)*, Sep. 2013, pp. 337–340.

[19] S. S. Mahmoud and I. Ahmad, "A green model for sustainable software engineering," *Int. J. Softw. Eng. Appl.*, vol. 7, no. 4, pp. 55–74, 2013.

[20] M. Dick, J. Drangmeister, E. Kern, and S. Naumann, "Green software engineering with agile methods," in *Proc. 2nd Int. Workshop Green Sustain. Softw. (GREENS)*, May 2013, pp. 78–85.

[21] S. Aguado, R. Alvarez, and R. Domingo, "Model of efficient and sustainable improvements in a lean production system through processes of environmental innovation," *J. Cleaner Prod.*, vol. 47, pp. 141–148, May 2013.

[22] K. Tate, *Sustainable Software Development: An Agile Perspective*. Reading, MA, USA: Addison-Wesley, 2005.

[23] R. Abdullah, S. Abdullah, J. Din, and M. Tee, "A systematic literature review of green software development in collaborative knowledge management environment," *Int. J. Adv. Comput. Technol.*, vol. 9, no. 11, p. 136, Oct. 2015.

[24] S. Sahay, "Global software alliances: The challenge of'standardization," *Scand. J. Inf. Syst.*, vol. 15, no. 1, p. 11, 2003.

[25] M. Niazi, N. Ikram, M. Bano, S. Imtiaz, and S. U. Khan, "Establishing trust in offshore software outsourcing relationships: An exploratory study using a systematic literature review," *IET Softw.*, vol. 7, no. 5, pp. 283–293, Oct. 2013.

[26] A.-M. Soderberg and L. Romani, "Interorganizational boundary spanning in global software development," in *Proc. 13th EIASM Workshop Int. Manage.*, 2015, pp. 32–41.

[27] A. Danait, "Agile offshore techniques-a case study," in *Proc. Agile Develop. Conf. (ADC)*, Jul. 2005, pp. 214–217.

[28] C. Young and H. Terashima, "How did we adapt agile processes to our distributed development?" in *Proc. Agile Conf.*, Aug. 2008, pp. 304–309.

[29] L. Ahuja and R. Priyadarshi, "Agile approaches towards global software engineering," in *Proc. 4th Int. Conf. Rel., INFOCOM Technol. Optim. (ICRITO) (Trends Future Directions)*, Sep. 2015, pp. 1–6.

[30] N. B. Moe, D. Cruzes, T. Dyba, and E. Mikkelsen, "Continuous software testing in a globally distributed project," in *Proc. IEEE 10th Int. Conf. Global Softw. Eng.*, Jul. 2015, pp. 130–134.

[31] S. Ray, "Green software engineering process: Moving towards sustainable software product design," *J. Global Res. Comput. Sci.*, vol. 4, no. 1, pp. 25–29, 2013.

[32] T. Dingsøyr, S. Nerur, V. Balijepally, and N. B. Moe, "A decade of agile methodologies: Towards explaining agile software development," *J. Syst. Softw.*, vol. 85, no. 6, pp. 1213–1221, Jun. 2012.

[33] R. J. Koontz and R. L. Nord, "Architecting for sustainable software delivery," Softw. Eng. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. FA8721-05-C-0003, 2012.

[34] M. Paasivaara and C. Lassenius, "Could global software development benefit from agile methods?" in *Proc. IEEE Int. Conf. Global Softw. Eng. (ICGSE)*, Oct. 2006, pp. 109–113.

[35] E. Kern, S. Naumann, and M. Dick, "Processes for green and sustainable software engineering," in *Green in Software Engineering*. New York, NY, USA: Springer, 2015, pp. 61–81.

[36] M. Pankowska, "Sustainable software: A study of software product sustainable development," in *Mechanism Design for Sustainability*. Berlin, Germany: Springer, 2013, pp. 265–281.

[37] J. Stammel, Z. Durdik, K. Krogmann, R. Weiss, and H. Koziolek, *Software Evolution for Industrial Automation Systems: Literature Overview*. Munich, Germany: KIT, Fakultät für Informatik, 2011.

[38] C.-Y. Hsieh and C.-T. Chen, "Patterns for continuous integration builds in cross-platform agile software development," *J. Inf. Sci. Eng.*, vol. 31, no. 3, pp. 897–924, 2015.

[39] J. Taina, "Good, bad, and beautiful software-in search of green software quality factors," *Cepis Upgrade*, vol. 12, no. 4, pp. 22–27, 2011.

[40] M. Kodali, "Traceability of requirements in scrum software development process," Ph.D. dissertation, School Innov. Des. Eng., Malardalen Univ., Västerås, Sweden, 2015.

[41] E. A. Altameem, "Impact of agile methodology on software development," *Comput. Inf. Sci.*, vol. 8, no. 2, p. 9, 2015.

[42] S. U. Jan, "Framework for the development of sustainable green software," Univ. Malakand, Chakdara, Pakistan, Tech. Rep. 133-221, 2019.

[43] O. A. Galán, J. L. Valdéz, H. F. Medina, G. A. Contreras, and J. L. Sumuano, "Proposal of a sustainable agile model for software development," *Development*, vol. 11, no. 1, pp. 1–8, 2020.

[44] P. Bozzelli, Q. Gu, and P. Lago, "A systematic literature review on green software metrics," Dept. Comput. Sci., VU Univ., Amsterdam, The Netherlands, Tech. Rep. 25-2013, 2013.

[45] B. Kitchenham and C. Charters, "Guidelines for performing systematic literature reviews in software engineering," EBSE, Keele Univ., Keele, U.K., Tech. Rep. EBSE-2007-01, Ver. 2.3, 2007.

[46] B. A. Kitchenham, D. Budgen, and P. Brereton, *Evidence-Based Software Engineering and Systematic Reviews*. vol. 4. Boca Raton, FL, USA: CRC Press, 2015.

[47] M. Niazi, S. Mahmood, M. Alshayeb, M. R. Riaz, K. Faisal, N. Cerpa, S. U. Khan, and I. Richardson, "Challenges of project management in global software development: A client-vendor analysis," *Inf. Softw. Technol.*, vol. 80, pp. 1–19, Dec. 2016.

[48] S. Ali and S. U. Khan, "Practices for implementation of the critical success factors in software outsourcing partnership from vendors' perspective: A literature review," *Proc. Pakistan Acad. Sci.*, vol. 53, no. 2, pp. 145–162, 2016.

[49] M. Bano and D. Zowghi, "A systematic review on the relationship between user involvement and system success," *Inf. Softw. Technol.*, vol. 58, pp. 148–169, Feb. 2015.

[50] T. R. D. Saputri and S.-W. Lee, "The application of machine learning in self-adaptive systems: A systematic literature review," *IEEE Access*, vol. 8, pp. 205948–205967, 2020.

[51] D. E. Perry, A. A. Porter, and L. G. Votta, "Empirical studies of software engineering: A roadmap," in *Proc. Conf. Future Softw. Eng.*, 2000, pp. 345–355.

[52] I. Nurdiani, R. Jabangwe, D. Smite, and D. Damian, "Risk identification and risk mitigation instruments for global software development: Systematic review and survey results," in *Proc. IEEE 6th Int. Conf. Global Softw. Eng. Workshop*, Aug. 2011, pp. 36–41.

[53] R. Rabiser, P. Grünbacher, and D. Dhungana, "Requirements for product derivation support: Results from a systematic literature review and an expert survey," *Inf. Softw. Technol.*, vol. 52, no. 3, pp. 324–346, Mar. 2010.

[54] M. A. Akbar, W. Naveed, A. A. Alsanad, L. Alsuwaidan, A. Alsanad, A. Gumaei, M. Shafiq, and M. T. Riaz, "Requirements change management challenges of global software development: An empirical investigation," *IEEE Access*, vol. 8, pp. 203070–203085, 2020.

[55] N. Rashid and S. U. Khan, "Using agile methods for the development of green and sustainable software: Success factors for GSD vendors," *J. Softw., Evol. Process*, vol. 30, no. 8, p. e1927, Aug. 2018.

[56] N. Rashid and S. U. Khan, "Agile practices for global software development vendors in the development of green and sustainable software," *J. Softw., Evol. Process*, vol. 30, no. 10, p. e1964, 2018.

[57] N. Rashid and S. U. Khan, "Developing green and sustainable software using agile methods in global software development: Risk factors for vendors," in *Proc. 11th Int. Conf. Eval. Novel Softw. Approaches Softw. Eng.*, 2016, pp. 247–253.

[58] R. Constantinescu and I. M. Iacob, "Capability maturity model integration," *J. Appl. Quant. Methods*, vol. 2, no. 1, p. 187, 2007.

[59] M. Niazi, D. Wilson, and D. Zowghi, "A maturity model for the implementation of software process improvement: An empirical study," *J. Syst. Softw.*, vol. 74, no. 2, pp. 155–172, Jan. 2005.

[60] S. U. Khan, "Software outsourcing vendors' readiness model (SOVRM)," Ph.D. dissertation, Dept. Comput. Sci., Keele Univ., Keele, U.K., 2011.

[61] N. Rashid and S. U. Khan, "Green agile maturity model for global software development vendors," *Sci. Int.*, vol. 26, no. 5, pp. 1–3, 2014.

[62] M. Niazi, W. David, and Z. Didar, "A framework for assisting the design of effective software process improvement implementation strategies," *J. Syst. Softw.*, vol. 78, no. 2, pp. 204–222, Mar. 2005.

[63] S. Ali and S. U. Khan, "Software outsourcing partnership model: An evaluation framework for vendor organizations," *J. Syst. Softw.*, vol. 117, pp. 402–425, Jul. 2016.

[64] R. K. Yin, *Case Study Research: Design and Methods*. Newbury Park, CA, USA: Sage, 2009.

[65] D. Cooper and P. Schindler, *Business Research Methods*, 7th ed. New York, NY, USA: McGraw-Hill, 2001.

[66] A. L. Peres and S. L. Meira, "Towards a framework that promotes integration between the UX design and SCRUM, aligned to CMMI," in *Proc. 10th Iberian Conf. Inf. Syst. Technol. (CISTI)*, Jun. 2015, pp. 1–4.

[67] M. K. Daskalantonakis, "Achieving higher SEI levels," *IEEE Softw.*, vol. 11, no. 4, pp. 17–24, Jul. 1994.

[68] S. Beecham, T. Hall, and A. Rainer, "Assessing Requirements Process Strengths and Weaknesses: A first step to prioritising requirements process implementation," Dept. Comput. Sci., Univ. Hertfordshire, Hatfield, U.K., Tech. Rep. 381, 2003.

[69] M. Niazi, K. A. Cox, and J. M. Verner, "An empirical study identifying high perceived value requirements engineering practices," in *Advances in Information Systems Development*. Springer, 2006, pp. 731–743.

[70] M. Niazi, "A comparative study of software process improvement implementation success factors," *J. Softw., Evol. Process*, vol. 27, no. 9, pp. 700–722, 2015.

[71] P. Legris, J. Ingham, and P. Collerette, "Why do people use information technology? A critical review of the technology acceptance model," *Inf. Manage.*, vol. 40, no. 3, pp. 191–204, Jan. 2003.

[72] P. Hodkinson and H. Hodkinson, "The strengths and limitations of case study research," in *Proc. Learn. Skills Develop. Agency Conf. Cambridge*, 2001, pp. 5–7.

[73] M. Niazi, "Do systematic literature reviews outperform informal literature reviews in the software engineering domain? An initial case study," *Arabian J. Sci. Eng.*, vol. 40, no. 3, pp. 845–855, Mar. 2015.

NASIR RASHID received the Ph.D. degree in computer science from the University of Malakand, Pakistan. He is currently an Assistant Professor with the Computer Science and IT Department, University of Malakand. His research interests include agile strategy and green software outsourcing.

SIFFAT ULLAH KHAN received the Ph.D. degree in computer science from Keele University, U.K., in 2011.

He was the Head of the Department of Software Engineering, University of Malakand, Pakistan, for three years, where he was also the Chairman of the Department of Computer Science and IT. He is currently an Associate Professor in computer science with the University of Malakand. He is also the Founder and the Leader of the Software Engineering Research Group, University of Malakand. He has successfully supervised ten M.Phil. and four Ph.D. Scholars. He has authored over 100 articles, so far, in well-reputed international conferences and journals. His research interests include software outsourcing, empirical software engineering, agile software development, systematic literature review, software metrics, cloud computing, requirements engineering, and green computing/IT. He received the Gold Medal (Dr. M. N. Azam Prize 2015) from the Pakistan Academy of Sciences in recognition of his research achievements in the field of computer (software).

HABIB ULLAH KHAN received the Ph.D. degree in management information systems from Leeds Beckett University, U.K. He is currently working as a Professor of management information systems with the Department of Accounting and Information Systems, College of Business and Economics, Qatar University, Qatar. He has nearly 20 years of industry, teaching, and research experience. His research interests include IT adoption, social media, Internet addiction, mobile commerce, computer mediated communication, IT outsourcing, big data, and IT security.

MUHAMMAD ILYAS received the Ph.D. degree in computer science from the University of Malakand, Pakistan. He is currently an Assistant Professor with the Computer Science and IT Department, University of Malakand. His research interests include software outsourcing, empirical software engineering, systematic literature review, cloud computing, requirements engineering, and green computing/IT.