IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 11 March 2024, accepted 28 March 2024, date of publication 2 April 2024, date of current version 19 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3384410

RESEARCH ARTICLE

Factors Affecting Agile Software Project Success

BURCU BINBOGA AND CIGDEM ALTIN GUMUSSOY
Department of Industrial Engineering, Istanbul Technical University, 34367 Istanbul, Türkiye

Corresponding author: Cigdem Altin Gumussoy (altinci@itu.edu.tr)

This work involved human subjects in its research. Approval of all ethical and experimental procedures and protocols was granted by the Ethics approval was obtained from Istanbul Technical University Social and Human Sciences Human Ethics Committee under Application NO. 262, date: 2022.06.16.

ABSTRACT Uncertainties and changes in software requirements have compelled companies to become more agile in software development projects. To keep pace with the agile environment, companies incorporate agile software development methodologies into their projects. However, the way that the agile methodology is implemented may determine the success of the project. This study aims to understand critical success factors affecting agile projects' success from the perspectives of agile practitioners. First, an extensive systematic literature review was conducted with the detailed examination of the Agile Manifesto, Agile Principles, and Scrum Guide to develop the Agile Software Project Success Model. Second, refinement of the factors was conducted through one-on-one meetings with six agile practitioners, and then a group meeting was conducted to reach a consensus on the model. Finally, using IBM Amos 20.0, data collected from 596 agile practitioners were used to understand the relationships defined in the research model. Results indicated that all the hypotheses except two of them were supported. Among the factors, customer factors and agile process factors are the stronger predictors of process efficiency, sustainable software product quality and stakeholder satisfaction compared to other factors. This study concludes with the theoretical and practical implications and recommendations for possible future studies.

INDEX TERMS Agile project management, critical success factors, agile project success, risk factors, structural equation modelling.

I. INTRODUCTION

The project management methodology used in software development plays a pivotal role in the overall success of a project. The methodology is generally determined according to the customers' requirements and the organizational context of the company. However, the ambiguity in software qualifications and quick changes in customer needs may require software development activities to be faster and more agile. Therefore, in recent years, agile software development methodologies have frequently been used to respond customer and business needs.

Agile software development encompasses a comprehensive array of frameworks and methodologies that are founded upon the fundamental principles and values outlined in the Agile Manifesto and Agile Principles [1]. The agile software development methodology promises frequent release cycles, continuous improvement, fast feedback to rapidly changing requirements, face-to-face conversation, and high-quality software [2], [3]. Thus, many companies have recently preferred to use agile software development methodologies instead of traditional plan-based ones. The recent Standish Group Chaos report indicates that agile project success rates are three times higher than those of waterfall projects and that waterfall projects' failure rates are two times higher than those of agile projects [4]. Although agile projects have higher success rates than traditional ones, understanding agile critical success factors (CSFs) affecting agile projects' success is important for the software development management. CSFs for projects refer to "characteristics, conditions, or variables that can have a significant impact on the success of the project when properly sustained, maintained and managed" [5] (p. 183). "The patterns of success and failure, often referred to as CSFs, are some

The associate editor coordinating the review of this manuscript and approving it for publication was Shih-Wei Lin.

of the best indicators of the lessons learned, adapted, and used by software industry today'' [6] (p. 2). When addressed properly, CSFs significantly increase the likelihood of project success [7].

Several studies have focused on the success, failure, and risk factors in agile software development projects, such as success factors for agile methodologies [6], [8], [9], [10], [11], [12], [13], [14], [15], [16], risk factors affecting agile project success [17], [18], and agile failure factors [8], [11]. Although these studies provide insights into the individual effects of success factors, risk, and failure factors on project success, the objective of this study is to develop a comprehensive agile software project success model that combines agile CSFs with agile project success measures. To ascertain the critical factors in agile software project management, a detailed literature review has been conducted. In contrast to previous studies, the Agile Manifesto, Agile Principles, and Scrum Guide were examined in detail to identify significant factors in the application of agile software development projects. Additionally, the factors were refined via individual and group meetings with six agile practitioners working in the software industry. Finally, Agile Software Project Success Model was developed and tested with a large-scale survey data collected from 596 agile practitioners. The developed model can be used by scrum masters, product owners, developers, and researchers as a reference model to understand the critical factors affecting agile project success measures such as process efficiency, product quality, and stakeholder satisfaction.

The subsequent sections of this paper are structured as follows. Section II presents the comprehensive review of current literature on CSFs, and project success measures. Section III presents research methodologies. Section IV explains the Agile Software Project Success Model, the agile success measures, and the agile CSFs. Section V presents data analysis and results. Section VI presents discussion of results. In Section VII, research limitations and future research directions are proposed, and in Section VIII, conclusions are stated.

II. LITERATURE REVIEW

In the literature review, firstly studies related with CSFs in agile project management were examined. There is a comprehensive study related to CSFs for agile projects [6], [8], [9], [10], [11], [12], [13], [14], [15], [16]. Chow and Cao [8] composed a preliminary list of CSFs for agile projects from the literature and then grouped these factors into five constructs: organization, people, processes, technical, and project. The effects of agile project success factors on agile project success were analyzed with data collected from 109 agile software projects in 25 different countries. Stankovic et al. [11] used the CSF framework presented by Chow and Cao [8] and analyzed the effects of CSFs. Ahimbisibwe et al. [12] extracted a total of 37 CSFs from the literature for software development projects and categorized these factors into three groups: customer, organizational, and team factors. Subsequently, they developed a contingency fit model of the CSFs to compare traditional project management methodologies with the agile ones. Ahimbisibwe et al. [14] used the contingency fit model presented by Ahimbisibwe et al. [12] to compare traditional project management and agile methodologies in terms of CSFs for outsourced software development projects. Garousi et al. [6] used the same contingency model to evaluate the relative importance of CSFs for the software project's success.

A review of the literature on CSFs reveals that, in some studies, the CSFs were limited to specific constructs, such as organizational and people-related factors [9]; people, process, and technology-related factors [13]; success factors to develop green and sustainable software using agile methodologies [15]; the effect of client involvement [19]; the quality-related factors on the agile development process [16]; and people-related factors [20]. Misra et al. [9] defined success factors associated with the adoption of agile software development practices, focusing on organizational and people-related factors. Silva and Santos [13] grouped CSFs for agile software projects into people, process, and technology constructs. Each construct was explained with a set of CSFs. Rashid and Khan [15] focused on the green and sustainable software development using agile methodologies for software development vendors operating on a global scale. Perera and Perera [19] investigated the effect of client involvement such as client communication level, client collaboration on agile project success. Arcos-Medina and Mauricio [16] conducted a systematic literature review on quality in the agile development process to determine critical success measures, quality attributes, and agile practices. Tam et al. [20] focused on people-related factors affecting the agile software development project success, such as customer involvement, training and learning, societal culture, team capability, and personal characteristics. Although these studies [6], [8], [9], [10], [11], [12], [13], [14], [15], [16], [19], [20] provide a comprehensive list of success factors for agile projects, we propose an Agile Software Project Success Model that explains the relationships between agile CSFs and software project success measures from a comprehensive perspective. Furthermore, in addition to the literature review, the Agile Manifesto, Agile Principles, and Scrum Guide were examined in detail to reveal possible factors not mentioned in the literature. The model and the factors were validated by agile practitioners working in the software development industry. Therefore, the current study contributes to the literature by constructing a model, while grouping agile critical success factors into customer factors, team factors, organizational factors, agile process factors, technical factors, and project factors (agile risk and failure factors), agile project success measures were grouped into process efficiency, sustainable software product quality, and stakeholder satisfaction. Furthermore, the relationships defined in the model were tested and validated by collecting data from 596 agile practitioners. The participants also come from many different countries, had different working models such as remote work, hybrid work, and office-based work and had diverse IT job titles including developers, business analysts, testers, and others, as well as different levels of experience.

Furthermore, in the development of complex systems, the lack of risk identification process is one of the primary reasons for project failure [21]. Therefore, risk and failure factors significantly affect the success of software development projects. In previous studies, risk factors and their effects on software development projects were examined [17], [18], [22], [23], [24]. The studies of Shrivastava and Rathod [17], [18] obtained a set of risk factors influencing the performance of distributed agile software development projects and developed a risk management framework. Chow and Cao [8] and Stankovic et al. [11] examined not only agile CSFs but also the main causes of software project failures. In the present study, agile project risk factors and failure factors are considered as possible determinants of project success measures together with customer factors, team factors, organizational factors, agile process factors, and technical factors.

III. RESEARCH METHODOLOGY

In this study, the systematic literature review (SLR) approach proposed by Kitchenham and Charters [25] was used to improve the quality of the literature review process and to standardize it. First, research questions related to the area of interest were formulated. Then, data sources, search criteria, and article selection process were applied to determine the CSFs, project success measures. The enhancement of the factors was conducted a meticulous process through one-on-one meetings with six practitioners well-versed in agile methodologies. Subsequently, a group meeting was convened to attain a unanimous agreement on the conceptual model. Ultimately, the dataset derived from 596 agile practitioners by survey was used, thereby elucidating the relationships delineated within the research model. The research methodology used in the current study is shown in Figure 1.

A. RESEARCH QUESTIONS

According to Kitchenham and Charters [25], "SLR is a systematic method for identifying, evaluating and interpreting all currently available research relevant to the particular research question(s) or area of interest." The research questions of the present study focused on determining and categorizing agile CSFs, project risk factors, project failure factors, and project success measures:

RQ1. What are the CSFs of agile software projects?

RQ2. What are the success measures of agile software projects?

RQ3. To which constructs do the CSFs belong?

RQ4. To which constructs do project success measures belong?

B. DATA SOURCES

The SLR was conducted using the following digital library sources: ScienceDirect, IEEE, Springer, Wiley, and Emerald. Academic journals and conference articles were considered in the selection process. It covered all related articles published from 2000 to 2021. The end date was set to October 29, 2021, for SLR. After SLR, the data collection phase of the survey spanned approximately one calendar year. Meanwhile, a continuous literature review endured, meticulously reading over, and analyzing publications from 2022 and 2023, thereby ensuring the incorporation of contemporary literature.

C. SEARCH CRITERIA

Having a search criterion for the SLR was essential for the next phases. The search string format was created using the main terms of the research questions. The search string used in this study was as follows: ((agile OR scrum) AND software) AND ("project success" OR "critical success factor" OR "project failure" OR "project risk").

D. ARTICLE SELECTION PROCESS

The search string was used in the article selection process. In the first step, 3702 articles were extracted from selected database sources. In the second step, the inclusion/exclusion criteria were applied by reading the titles, abstracts, introductions, and conclusions. The inclusion criteria were as follows:

- Articles written in English.
- Articles published between 2000 and 2021.
- Articles available online.
- Articles that mention/describe CSFs in agile software projects or in a general context.
- Articles related to academia and industry.
- Articles that include an empirical, theoretical, or expert opinion on success factors and success measures in software development

The exclusion criteria were as follows:

- Articles not related to software development (e.g., application of Kanban boards in the manufacturing industry).
- Short articles (less than six pages).

By applying these criteria and selection based on title, abstract, introduction and conclusion, the number of articles was reduced to 133. After thoroughly reading these 133 articles, 27 articles were shortlisted. Six articles were added based on internet searches and the references of shortlisted articles. Finally, 33 articles were identified via the article selection process, which are presented in Appendix A (SLR1-SLR33). Table 1 presents the distribution of the selected articles.

IV. AGILE SOFTWARE PROJECT SUCCESS MODEL

The Agile Software Project Success Model was constructed based on studies related with agile CSFs, and project success measures. Furthermore, the Agile Manifesto, Agile Principles, and Scrum Guide were examined in detail to have a better comprehension of the usage of agile methodologies.

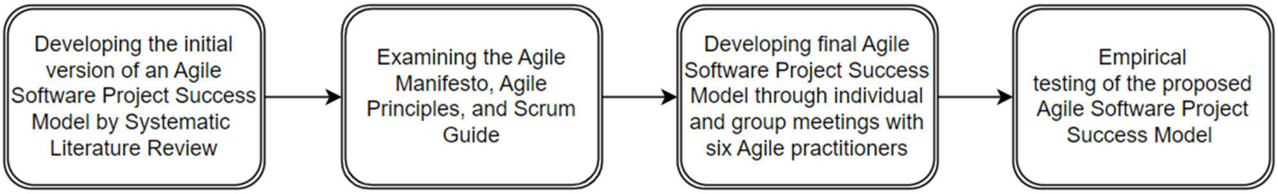


FIGURE 1. The research methodology.

TABLE 1. Distribution of the selected articles.

Source	Potentially eligible articles	Studies after primary elimination	Selected articles	Ratio (%)
Science Direct	1050	52	11	33.33
Emerald	1034	30	4	12.12
IEEE	69	14	5	15.15
Springer	1261	19	5	15.15
Wiley	288	20	2	6.06
Snowballing			6	18.18
Total	3702	133	33	100

TABLE 2. The demographic profile of agile practitioners.

Job Role	Industry	Work Experience	Agile Methodologies
Scrum Master	Banking Industry	10 -15 years	Scrum, Kanban
Scrum Master	Aviation Industry	10 -15 years	Scrum
Software Developer	Banking Industry	15 years or above	Scrum
Software Developer	Telecommunication	10 -15 years	Scrum
Product Owner	Consulting Industry	5 -10 years	Scrum, Kanban XP, Lean
Business Analyst	Stock Exchange	10 -15 years	Scrum

The Agile Manifesto was used to emphasize agile values, such as responding to change, customer collaboration, working software, and valuing individuals and interactions. Additionally, Agile Principles, events, artifacts, and values were examined to add additional constructs to a conceptual model. Thus, agile CSFs, and project success measures of the Agile Software Project Success Model were identified.

In the second step, one-on-one meetings with agile practitioners were conducted to have a discuss about the Agile Software Project Success Model. Participants included agile practitioners who had worked as a scrum master, development team members, and product owner. Purposive sampling was used for the selection of participants, wherein, the sample complies with the criterion that the subjects are agile practitioners, who have worked on or are currently working on agile projects. As suggested by in the study of Tam et al. [20], six agile practitioners were selected for meetings. The demographic profile of agile practitioners is shown in Table 2. At the beginning of the meeting, general information about the agile practices used by the participants was obtained. All the agile practitioners applied agile practices such as agile events, an efficient software testing process, refactoring activities, a product/sprint backlog, an agile-friendly progress tracking/controlling mechanism with burndown charts, and a sprint progress dashboard. Then, the agile CSFs, and success measures identified through the prior literature were discussed in detail. The agile practitioners' comments about the literature review results were incorporated into the explanation of the agile CSFs, and success measures. With the recommendations provided by the practitioners, minor wording changes to the CSFs, and success measures were made for better comprehension.

In the last step, at the group meeting, the group of six agile practitioners finalized the agile CSFs, and project success measures in the Agile Software Project Success Model. The agile practitioners emphasized an agile-friendly progress tracking/controlling mechanism, simplicity, self-organized team members, project manager's experience, customer's experience in its business domain, and urgency. In addition, functional suitability, usability, maintainability, portability (transferability), performance efficiency, compatibility, software reliability, security, and upper management satisfaction were highly valued by agile practitioners in the group meeting. Finally, the Agile Software Project Success Model was constructed, as shown in Fig. 2.

A. AGILE CRITICAL SUCCESS MEASURES FOR SOFTWARE DEVELOPMENT PROJECTS

The Project Management Institute (PMI) defines project success in its guideline as the completion of the project within the agreed time, budget, and scope [26]. These three constructs are referred to as the "iron triangle." Chow and Cao [8] added quality (i.e., delivering good project outcomes) to the constructs defined by the PMI. Ahimbisibwe et al. [12] improved the description of project success and included the reliability, ease of use, overall quality, flexibility, and functionality of the software delivered, team satisfaction, upper management satisfaction, and user satisfaction. In this paper, the agile project success measures were determined according to the literature review, Agile Manifesto, Agile Principles, Scrum Guide, ISO/IEC 25010: 2011 quality model [27], and comments obtained from agile practitioners

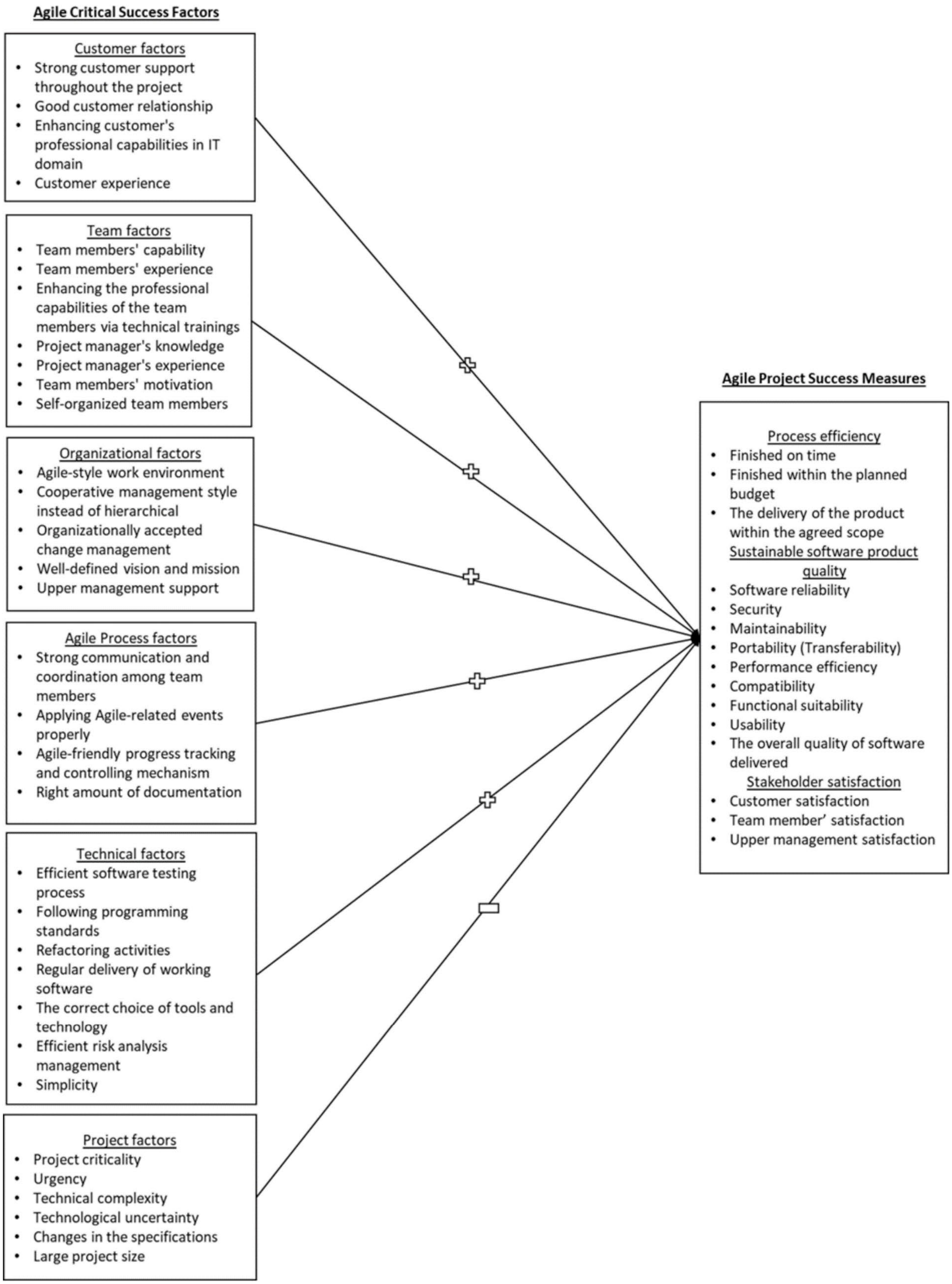


FIGURE 2. Agile software project success model.

via one-on-one meetings and the group meeting. The agile success measures were categorized into three constructs: process efficiency, sustainable software product quality, and stakeholder satisfaction. The agile project success measures and the references related with these measures are presented in Table 3.

TABLE 3. Agile project success measures.

Constructs	Agile project success measures
Process efficiency	Finished on time [6], [8], [11-12], [14], [28-33]
	Finished within the planned budget [6], [8], [11-12], [14], [28-33]
	The delivery of the product within the agreed scope [11-12], [14], [17], [29-32]
Sustainable software product quality	Software reliability [6], [12], [14], [16]
	Security [6], [16]
	Maintainability [6], [16]
	Portability [6], [16]
	Performance efficiency [6], [16]
	Compatibility [6], [16]
	Functional suitability [6], [12], [14], [33-34]
	Usability [6], [12], [14], [16]
	The overall quality of software delivered [6], [8], [11-12], [14], [29-31], [33], [35]
Stakeholder satisfaction	Customer satisfaction [6], [12], [14], [28], [30-33]
	Team members' satisfaction [6], [12], [14], [32-33]
	Upper management satisfaction [6], [12], [14]

1) PROCESS EFFICIENCY

In agile project management, the confluence of time, budget, and scope is essential for assessing success. Navigating iterative cycles on time, managing resources rigorously within budget, and remaining responsive to stakeholders within the agreed scope constitute the cornerstones of agile project success. In the current study, process efficiency is explained by three characteristics: finished on time, finished within the planned budget, and the delivery of the product within the agreed scope.

Finished on time refers to completing the project on time by adhering to the project schedule. According to the literature review, software delivery within the determined cost, time, and budget is the most cited project success measure [6], [8], [11], [12], [30], [36].

Finished within the planned budget. Every project begins with a plan and a budget. However, in many cases, delivering the project within the planned budget is a considerable challenge. According to a recent report of CHAOS 2020: Beyond Infinity, only 35% of projects were fully successful regarding time and budget [4].

The delivery of the product within the agreed scope refers to completing the project within the predetermined scope. The project scope consists of predetermined tasks required to achieve the project's goals. Product delivery within the agreed scope is considered as one of the success measures [6], [8], [11], [12], [31], [32].

2) SUSTAINABLE SOFTWARE PRODUCT QUALITY

Sustainability of a software product is regarded as a non-functional requirement in the software engineering literature, and quality attributes of software products have been associated with sustainability [37]. In the studies of Condori-Fernandez and Lago [38], and Condori-Fernandez et al. [39], the software sustainability is explained by software product quality attributes defined by ISO/IEC 25010: 2011 quality model [27]. In the current study, sustainability of software product quality is defined by eight characteristics: reliability, security, maintainability, portability, performance efficiency, compatibility, functional suitability, and usability.

The ISO/IEC 25010: 2011 quality model defines ***software reliability*** as the "degree to which a system, product or component performs specified functions under specified conditions for a specified period of time" [27]. Software reliability helps to predict the number of defects or faults of a software product [40]. Reliability is primarily related to the social and economic dimensions of software sustainability, such as improving the software usage life span and reducing support and development costs [37]. Garousi et al. [6] found that the reliability of the delivered software/service is an essential project success measure. In the group meeting, all the participants agreed that software reliability is one of the project success measures.

System security refers to the ability of the product to protect itself from intentional or accidental intrusion (i.e., the ability to resist attack). The system's ability to resist unauthorized access, whether malicious or accidental, defines software product's success. Few papers mentioned software security as a project success measure [6]. A software developer with 10 years of experience said, "We should be aware of the possible types of attacks that could be encountered against infrastructure or services. Furthermore, we should incorporate the quality attribute of security into software development phases for sustainable software product."

Maintainability is "the degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers" [27]. There are three types of maintenance: corrective, adaptive, and perfective. Corrective maintenance aims to address system deficiencies in processing, performance, or implementation. Regarding adaptive maintenance, the systems must be adaptable to changes, for instance, upgrades of the operating system. Perfective maintenance involves perfecting the system regarding performance, processing efficiency, or maintainability [41]. Reducing support and maintenance costs, improving the software usage life span, and enabling the software to achieve societal needs constantly enhance software sustainability [37].

Portability is "the degree of effectiveness and efficiency with which a system, product or component can be transferred from one hardware, software or other operational or usage environment to another" [27]. If an application can be migrated to multiple platforms over its lifetime, its useful life is likely to be extended, and its user base may be expanded [37], [42]. Portability extends the software and hardware lifetime, reduces waste, improves the adaptability of software usage, and increases loyalty and satisfaction of customers [37]. In the group meeting, one of the experienced software developers said, "Some programming languages, such as C, are relatively portable. Because of the portability of C language program, some software developers prefer rewriting and recompiling their programs in C language."

Performance efficiency is "the degree to which a component or system uses time, resources and capacity when accomplishing its designated functions" [43]. It has three sub-characteristics: time behavior, resource utilization, and capacity. Performance efficiency reduces energy consumption, reduces e-waste, and increases software productivity [37], [38]. An experienced software developer stated, "To determine the system performance, we can observe how fast software product responds to user input. Also, when the system reaches its maximum limit, such as the number of concurrent users in the system, we can observe how the system reacts and how the system performs its functions."

Compatibility is "the degree to which a product, system or component can exchange information with other products, systems or components and/or perform its functions, while sharing the same hardware or software environment" [27] (e.g., the compatibility of software with different operating systems such as Unix, Windows, and macOS). Compatibility minimizes investment and development costs while also improving user communication and information exchange flexibility [37], [38]. An experienced scrum master said, "Compatibility of the system gives broad information about product quality. For example, when developing a mobile application, whether it is compatible with mobile platforms such as iOS, Android, and others provides information about the quality of your product. Similarly, if your website works without problem in different browsers such as Google Chrome, Firefox, etc., it means your website has cross-browser compatibility."

Functional suitability is "the degree to which a product or system provides functions that meet stated and implied needs when used under specified conditions" [27]. It is divided into three subcategories: functional completeness, functional correctness, and functional appropriateness. Functional suitability enables the long-term usage and reduces maintenance costs [37]. In the studies of Ahimbisibwe et al. [12], [14] and Garousi et al. [6], functional suitability was identified as a project success measure. During the one-on-one meetings, practitioners emphasized that after releasing the product, they are glad to hear positive responses from the customer about the functions of the product.

Usability is "the degree to which a product or system can be used by specified users to achieve specified goals with effectiveness, efficiency, and satisfaction in a specified context of use" [27]. Design of a product aligned with the user requirements results in ease of system usages and user satisfaction eventually. Garousi et al. [6] considered usability a project success measure. During the one-on-one meetings, practitioners emphasized that to ensure the usability of the product, the following questions should have a positive response: Is it easy to learn how the product works? Is the product easy to use? Can different types of people use the product?

The Overall Quality of Software Delivered: In the project management triangle, the project's quality is constrained by time, budget, and scope. However, system quality represents a multifaceted and multidimensional construct that affects the project and product lifecycle [44]. According to the inspiring previous literature, the overall quality of software delivered is the most cited project success measure. In numerous studies [8], [11], [29], [30], [33], [44], [45], the "overall quality of software delivered" was identified as one of the project success measures.

3) STAKEHOLDER SATISFACTION

The Agile Manifesto states, "Customer collaboration over contract negotiation" [46]. One of the most important goals of agile software development is to achieve ***customer satisfaction*** through continuous software delivery. If the system is useful and trustworthy, customer satisfaction will be prolonged [38]. According to one-on-one meetings and the literature [9], [30], [32], customer satisfaction is one of the most important project success measures.

Team Satisfaction: According to the literature [32], [33], team members' satisfaction is considered as one of the project success measures. The Agile Principle states, "Build projects around motivated individuals. Give them the environment and the support they need. Trust them to get the job done" [47].

According to one-on-one meetings with the agile practitioners and the literature (e.g., [6], [12], [14]), ***upper management satisfaction*** is one of the important agile project success measures. It is also critical to provide upper management satisfaction to gain their support for future projects.

B. AGILE CSFS FOR SOFTWARE DEVELOPMENT PROJECTS

The agile CSFs, along with the corresponding references and constructs, are presented in Table 4. The constructs are defined as the customer factors, team factors, organizational factors, agile process factors, technical factors, and project factors.

1) CUSTOMER FACTORS

Agile team members desire ***strong customer support*** throughout the project. Stankovic et al. [11] stated that having customer support contributes to the success of agile software development projects, and a lack of customer commitment was identified as a top 10 risk factor in software development projects [22]. Agile practitioners emphasized that customer commitment helps in completing projects within the agreed time. This view was confirmed by the study of Yetton et al. [57], who reported that customer support increases the likelihood of on-time project completion.

A good customer relationship is important for agile software development projects. The Agile Principles emphasize the value of good customer relationships, stating that

TABLE 4. Agile CSFS, along with the corresponding references and constructs.

Constructs	CSFs
Customer factors	Strong customer support throughout the project [6], [8-9], [11-14], [16], [19-20], [34], [44], [48] Good customer relationship [8], [11], [16], [19], [49] Enhancing customer's professional capabilities in IT domain [6], [12], [14], [16] Customer experience in its business domain [6], [12], [14]
Team factors	Team members' capability [8-9], [11], [13], [16], [20], [44], [48-50] Team members' experience with software development [6], [8], [11-12], [14], [16], [32-33], [44], [48] Enhancing the professional capabilities of the team members via technical training [8-11], [13], [49], [51] Project manager's knowledge [8], [11], [16], [29] Project manager's experience [16], [24], [48] Team members' motivation [8], [11], [13], [16], [50-51] Self-organized team members [8], [11]
Organizational factors	Agile-style work environment [8-9], [11], [16], [33] Cooperative management style instead of hierarchical [6], [8], [11-12], [14], [16], [49] Organizationally accepted change management [6], [10], [12], [14-16], [34], [44], [48], [52] Well-defined vision and mission [6], [10], [12], [14], [16] Upper management support [6], [8], [10-14], [16], [33], [44], [48-49]
Agile process factors	Strong communication and coordination among team members [6], [8-9], [11-15], [34], [51-52] Applying agile-related events properly [8], [11], [16], [32], [34-35], [49], [52-53] Agile-friendly progress tracking and controlling mechanism [6], [9], [12], [14], [35], [48], [53] Right amount of documentation [8], [11], [15-16], [35], [52], [54]
Technical factors	Efficient software testing process [8], [11], [13], [16], [34], [50] Following programming standards [8], [11], [13], [16] Refactoring activities [8], [11], [15-16], [34-35] Regular delivery of working software [8], [11], [15-16], [35], [50] The correct choice of tools and technology [8], [10-11], [44] Efficient risk analysis management [8], [11], [16], [44], [48] Simplicity [8], [11], [13], [15-16]
Project factors	Project criticality [6], [8], [11-12], [14], [55] Urgency [6], [12], [55] Technical complexity [6], [12], [14], [22-24], [28], [55] Technological uncertainty [6], [12], [14], [22-23], [28], [33], [55-56] Changes in the specifications [6], [12], [14], [16], [22-23], [49] Large project size [6], [12], [14], [17-18], [23]

"Business people and developers must work together daily throughout the project" [47]. The relationship between customers and software development team members is important for project success [8], [11], [49], [54].

Enhancing the customer's professional capabilities in the IT domain. According to Misra et al. [9], openness to share information and lifelong learning of project team members and customers increase the likelihood of agile software development practices being successful. Furthermore, technical knowledge of customers increases their level of interest in the software development process and their willingness to support development team members. This CSF is also supported by the literature [6], [9], [12], [14] in which customer training and education are considered as success factors for software development projects.

The customer's experience in its own domain is important for completing the project on planned time and within the agreed budget. Customers that are experienced in the relevant domain may precisely define the requirements of the product and counteract unpredictable changes. One of the experienced software developers in the group meeting mentioned that "A customer who is experienced in his own business domain prevents us from wasting too much time on details, by accurately expressing business requirements. Experienced customers provide accurate feedback throughout the entire software process and help to complete the project within the planned budget. User acceptance testing process is very important in the agile software projects, and experienced customers help us to develop a more successful software product thanks to rigorous user acceptance testing." Furthermore, previous studies confirmed the positive relationship between customer experience and project success [6], [12], [14].

Thus, the hypotheses are given as follows:

H1: Customer factors positively affect the process efficiency.

H2: Customers factors positively affect the sustainable software product quality.

H3: Customers factors positively affect the stakeholder satisfaction.

2) TEAM FACTORS

The project team should comprise people having a sufficient capability, regarding technical, managerial, and vocational skills, to achieve project success. The importance of ***team members' capabilities*** for project success has been confirmed by several studies [8], [44], [48], [49], [50], [54]. Additionally, team members' incapability (e.g., insufficient vocational skills) is mentioned as a risk factor for organizations [22].

The literature also reveals that there is a relationship between ***team members' experience*** with software development and software project success [8], [10], [44], [54], [58], [59]. Successful projects require dedicated staff with experience and skill during product release [44]. Agile methodologies need experienced team members who adapt easily to changes.

Enhancing team members' professional capabilities via technical training increases the likelihood of software project success [8], [48]. Formal or informal technical training is essential for team members because of rapid technological changes. Agile methodologies emphasize agility and continuous learning. Therefore, informal training is preferred over cumbersome and long-term training. The adoption of agile methodologies entails the dissemination of tacit knowledge among the team members via strong communication [9].

In the study of Stankovic et al. [11], ***project manager knowledge*** was classified under the team capability, as the manager is considered as one of the agile team members. Project manager's competence and knowledge are considered as success factors for software development projects [8], [11], [29], [48], [54]. One of the experienced software developers underlined the importance of project manager knowledge in a one-on-one meeting: "It is not enough for the project manager to have only technical knowledge. The project manager should be familiar with the organization's policies and practices. The project manager should, in fact, follow the project processes from beginning to end and have sufficient technical and domain knowledge to foresee procedures that are likely to cause bottlenecks. The project manager should motivate and direct team members with his knowledge."

The project manager's experience is crucial in software development projects. A lack of knowledge and experience may cause project failure [29]. For a successful project, managers should be able to handle stress, manage people, and commit time for the project and should have problem-solving and communication skills in the relevant domain [54].

One of the Agile Principles is ***team members' motivation***. "Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done" [47]. Motivated team members influence agile software projects' success [8], [11]. Similarly, Fatema and Sakib [51] reported that working with motivated people enhances team effectiveness. One practitioner said that "The software industry is very stressful by nature; therefore, the motivation of the employees should be high to work efficiently and to finish tasks within the planned time. The upper management and the human resources department should take responsibility for "soft factors" such as communication between the team members and motivation among employees. They should find a better way to increase employee motivation in place of just giving promotions or salary arrangements."

Self-organized team members are important according to the Agile Principles, "The best architectures, requirements, and designs emerge from self-organizing teams" [47]. There is a pull system among self-organized team members instead of a push system. The team members decide what is the best for their work, rather than ordering tasks outside the team. They do not need command and control relationships as in the waterfall approach [60]. Coherent, self-organizing teamwork is considered a significant CSF that increases productivity and encourages learning [8], [11].

Thus, the hypotheses are given as follows:

H4: Team factors positively affect the process efficiency.

H5: Team factors positively affect the sustainable software product quality.

H6: Team factors positively affect the stakeholder satisfaction.

3) ORGANIZATIONAL FACTORS

Agile-Style Work Environment: The Agile Manifesto states, "The most efficient and effective method of conveying information to and within a development team is face-to-face conversation" [47]. Numerous articles mentioned co-located team members as a success factor for agile software development projects [8], [11], [58]. With the COVID-19 pandemic, the working style of companies has changed. Many companies adopt a hybrid work model. Integrating both working in an office and working in remote settings has the potential to heighten employee motivation and engagement to an organization [53].

Cooperative Management Style Instead of Hierarchical: Organizations that apply agile software development should adapt quickly to changes [9]. A cooperative management style, rather than a bureaucratic management style, should be used to create an organization that embraces change. Chow and Cao [8] and Stankovic et al. [11] also proposed a cooperative management style as a CSF for agile software development projects.

Organizationally Accepted Change Management: The Agile Principle states, "Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage" [47]. The organizationally accepted change management strategy may induce development team members to adapt to changes easily. In many studies, change management was considered as a project success factor [10], [15], [44], [48].

Well-Defined Vision and Mission: The vision guides the future of the project and the company, and the mission describes the aim of the company. Yeo [56] found that ambiguous business needs and an unclear vision are among the top five failure factors. A scrum master with 15 years of experience said, "When an employee realizes that the company's mission is in line with his own, that employee tries to perform better, and employee commitment to the company increases. Furthermore, a visionary software developer chooses the right tools and technology before the project starts. This reduces the potential financial losses and the complexity that team members must deal with."

The literature suggests that ***upper management support*** is a success factor for software development projects [6], [10], [12], [33], [48], [49]. Conversely, a lack of upper management support is a risk factor [22], [51], [61], [62]. Projects cannot be completed successfully unless the software development team has upper management support [44].

Thus, the hypotheses are given as follows:

H7: Organizational factors positively affect the process efficiency.

H8: Organizational factors positively affect the sustainable software product quality.

H9: Organizational factors positively affect the stakeholder satisfaction.

4) AGILE PROCESS FACTORS

Strong communication and coordination among team members enhance information sharing and collaboration within team members and help to prevent team conflicts [12]. Their positive effect on project success has been confirmed by several studies [8], [11], [15], [51], [63]. Furthermore, the Agile Manifesto emphasizes communication and coordination among team members, stating, "Individuals and interactions over processes and tools" [46]. Scrum events and pair programming in agile processes enhance communication and coordination and lead agile team to have a better plan and software design, better code, and better debugging [64].

Applying Agile-Related Events Properly: There are five main events in the agile-style software development process: sprint, sprint planning, daily meetings, sprint review, and sprint retrospective. Each sprint can be considered a project with a duration no longer than one month [65]. Agile estimation techniques are used during sprint planning. Planning poker, t-shirt size estimation, dot voting, and the bucket system are agile estimation techniques [66]. Daily meetings support strong communication among users and developers [8], [11]. Mann and Maurer [67] reported that daily meetings keep customers updated and prevent customer confusion regarding the software development. Similarly, Cohn and Ford [68] highlighted the importance of daily meetings. A sprint review is a type of meeting held at the end of each sprint. The team members present their work results to key stakeholders and discuss progress toward the product goal. The sprint retrospective meetings help team members to understand what was done right/wrong and what can be done better in the next sprint. This improves the product quality and productivity [69]. All the participants in the group meeting emphasized that applying agile-related events properly increases efficiency and productivity.

An agile-friendly progress tracking/controlling mechanism ensures that the project is completed on schedule and within the planned budget. Scrum artifacts include the product backlog, sprint backlog, and increment for tracking and controlling the project. The product owner and agile team members must know the progress of each artifact. The progress of each artifact can be measured: "For the Product Backlog it is the Product Goal. For the Sprint Backlog it is the Sprint Goal. For the Increment it is the Definition of Done" [65].

Right Amount of Documentation: The Agile Manifesto states, "Working software over comprehensive documentation" [46]. Agile methodologies require less documentation than traditional project management methodologies. However, in one-on-one meetings, agile practitioners stated that during the COVID-19 pandemic period, remote working conditions necessitated more documentation than before, because tacit knowledge transfer was difficult. They noted that it would be helpful to create other documents rather than only using the code to keep pace with changing requirements.

Thus, the hypotheses are given as follows:

H10: Agile process factors positively affect the process efficiency.

H11: Agile process factors positively affect the sustainable software product quality.

H12: Agile process factors positively affect the stakeholder satisfaction.

5) TECHNICAL FACTORS

An Efficient Software Testing Process: Inadequate software testing may cause software projects to fail [50] and is a risk factor for agile projects [17]. Therefore, efficient software testing before release and deployment of the product is crucial. The agile testing quadrants describe different types of testing approaches [70]. Quadrant 1 covers unit/component tests. Component testing is generally executed by the software developer who wrote the code to reduce risks and detect and prevent defects in the component. Quadrant 2 includes business-facing tests such as functional tests, story tests, prototypes, examples, and simulations. Quadrant 3 includes tests to determine whether the working software satisfies business requirements. User acceptance testing, usability testing, and exploratory testing are such examples. Quadrant 4 covers non-functional tests (e.g., performance testing and load testing) to evaluate non-functional requirements of a system or component [43].

Following Programming Standards: Development team members should follow standardization and best practices. Following a standard architecture [36] and following well-defined coding standards [8], [11], [13] are CSFs. Following coding standards help development teams to develop simple software programs. Programming standards may serve as a guideline for the development team members. When coding standards are followed, software product maintenance is easy because developers can understand code and modify it at any time.

Refactoring activity is an agile practice and involves improving the code structure and the software quality [71]. Code refactoring activities aim to improve non-functional attributes, such as the design, structure, and implementation of software, while maintaining product functionality. Refactoring activities are intended to guarantee the attainment of optimal outcomes and accommodate any modifications in project requirements [8], [11]. The effect of refactoring activities on agile project success is confirmed by the literature [8], [11], [15].

Agile methodologies focus on ***regular delivery of working software*** for agile projects' success [8], [11]. The delivery strategy is mentioned in the first and third Agile Principles: "continuous delivery of valuable software" and "deliver working software frequently" [47]. Without continuous delivery of valuable products, code integration and testing activities become more difficult [72]. Regular release dates with short iterations, continuous integration, and suitable testing activities can prevent risks in system release management and deployment [17].

The Correct Choice of Tools and Technology: Agile Principles state that development team members should focus on having the right technology and design to develop software with a rapid response capability [10]. The correct choice of tools and technology was identified as an agile CSF in several studies [8], [11]. It is important for the project team members to employ appropriate technologies, and tools for agile practices, such as tools facilitating iterative development, and processes supporting refactoring activities [8].

Efficient Risk Analysis Management: The evaluation of risks through risk analysis involves quantifying their frequency of occurrence and the magnitude of potential losses [73]. Risk management involves "coordinated activities to direct and control an organization with regard to risk" [74] and risk analysis assessments. Risk management reduces uncertainty and increases the likelihood of software development project success [75]. It is important to evaluate risks, measure their probabilities of occurrence, and determine their potential effects on the project [76]. Use of sprints in agile methodologies supports risk management because sprints enhance predictability through regular inspection and adaptation, enabling progress towards the product goal [65]. Project risk analysis is a success factor for agile software development projects [8], [11].

The tenth principle of the Agile Manifesto is "***Simplicity***– the art of maximizing the amount of work not done – is essential" [47]. Simplicity refers to keeping the design and code as simple as possible. In a simple design, duplicate code is avoided [77]. Team members try to develop a software product that is easy to understand and satisfies requirements [15]. Examples of simplicity include working software over comprehensive documentation (Agile Manifesto), use of the Gherkin language for a user story (agile practice), backlog refinement (agile practice), and keeping the code and design simple via refactoring and pair programming (agile practice). Pursuing a simple design increases the likelihood of agile project success [8], [11], [15].

Thus, the hypotheses are given as follows:

H13: Technical factors positively affect the process efficiency.

H14: Technical factors positively affect the sustainable software product quality.

H15: Technical factors positively affect the stakeholder satisfaction.

6) PROJECT FACTORS

A risk factor is delineated as a condition that may influence the successful completion of a software development project [78]. Through automated testing, usage of short iterations, embracing continuous integration, prioritizing feature-driven development, and various other procedural approaches, agile methodologies strive to mitigate software development risks. However, the persistent challenge of project failure or the possibility of exceeding budget and schedule remains an ongoing concern [79]. Herein, risk factors and failure factors are denoted as project factors, in accordance with the study of Ahimbisibwe et al. [12]. Six project factors were identified through the literature and meetings with the agile practitioners, as shown in Table 4: project criticality, urgency, technical complexity, technological uncertainty, changes in the specifications, and large project size.

The project criticality is an essential driver for many decisions. For a more critical project, it is more important to mitigate risk by using appropriate techniques, tools, and methods [80]. In several studies, project criticality was identified as a factor affecting project success [6], [12], [55]. Test driven development, right amount of documentation, pair programming, refactoring activities, appropriate technical training, following coding standards, and good communication between customer and team members are important activities for mitigating the risks associated with project criticality.

Urgency influences the relationships between CSFs and the attainment of project success [6], [12], [55]. A critical project with high urgency may reduce motivation among team members. Furthermore, in the group meeting, all practitioners stated that urgency places pressure on team members while they make a decision or handle a problem. Applying agile-related events properly, strong customer support and strong communication and coordination among team members may be helpful to reduce the possible risks of urgency.

Technical complexity is associated with the number of components developed, the use of new technology, and the difficulty of integration. The complexity of information technology systems influences project success and the quality of products [6], [24]. The successful completion of a project may be affected if its complexity is not sufficiently evaluated and addressed [81]. Technical complexity requires knowledgeable team members and as the level of technical complexity increases, team members' degree of control over the project may decrease. With the escalating levels of intricacy and interconnectedness exhibited by evolving systems, agile methodologies become more useful [82]. Some agile practices may be useful to deal with the technical complexity, such as pair programming, refactoring activities, following standards, and applying agile-related events properly.

Technological uncertainty is an important risk factor for software development processes. It is generally because of a lack of experience and technical knowledge of team members. Ahimbisibwe et al. [12] found that technological uncertainty affects the strength and/or direction of the relationship between CSFs and software project success measures. Appropriate technical trainings, more experienced team members, and retrospective meetings may be effective activities to manage the risks of technological uncertainty.

Changes in the Specifications: Scope creep stands out as an important factor in the failure of software development projects. Effectively handling and overseeing the elements of change within a project, specifically pertaining to the project's scope, emerges as a fundamental factor for project success [83]. Although agile methodologies accept changes in requirements, changes may cause deviations in the project scope, time, and budget and considered as a risk factor for software project management [22], [23] and project success [6], [12]. To reduce the potential risks arising from the frequent changes in the specifications, agile team may apply test driven development and refactoring activities to monitor project progress.

Large Project Size: An agile team typically consists of no more than 10 members. Studies indicated that team size growth is a risk factor for project success, particularly for distributed agile projects [17], [18]. The agile methodologies require strong communication among team members. As the number of team member increases, communication and coordination problems may occur. Furthermore, tacit knowledge transfer among team members may be difficult.

Thus, the hypotheses are given as follows:

H16: Project factors negatively affect the process efficiency.

H17: Project factors negatively affect the sustainable software product quality.

H18: Project factors negatively affect the stakeholder satisfaction.

V. EMPIRICAL ANALYSIS OF AGILE PROJECT SUCCESS MODEL

A survey questionnaire was prepared to test Agile Software Project Success Model, shown in Figure 2. Each construct in the model was defined with several items from the literature [6], [11], [14], [27], [44]. In the questionnaire, a total of 48 items that measure agile CSFs and agile project success measures was prepared and all the items are given in Appendix B. The final questionnaire includes two sections. The initial section included questions related with the demographics of the respondents (e.g., regarding the agile methodologies used, industry, the work model (onsite, remote, or hybrid), working country and job responsibility in the project (e.g., scrum master, product owner)). In the second section, the perception of the respondents about CSFs and software project success measures were asked on a seven-point Likert scale, with 1 indicating strongly disagree to 7 indicating strongly agree.

The target population for the current study includes the agile practitioners. To initiate this study, ethics approval was obtained from Istanbul Technical University Social and Human Sciences Human Ethics Committee (project number: 262, date: 2022.06.16). Before the application of large-scale survey, a pilot test was conducted with 20 agile practitioners to finalize the survey. After finalizing the questionnaire, a total of 596 questionnaires were collected from agile practitioners.

According to the demographic profiles, a large portion which is 73.32%, of the respondents are working in Turkey. The respondents working in the industry of banking and the stock exchange comprises nearly half of the respondents. Furthermore, a high portion (68.12%) of the respondents use scrum as an agile methodology. The details of the demographic profiles are given in Table 5.

In the analysis of the collected data, first explanatory factor analysis was performed to reveal the factor structure of the model. Then structural equation modeling approach with a two-step procedure was applied. The confirmatory factor analysis was performed to validate the model and the relationships defined in the Agile Software Project Success Model was tested with structural model.

A. EXPLORATORY FACTOR ANALYSIS

An exploratory factor analysis (EFA) with Principal component analysis and Varimax rotation was performed to determine the underlying factor structure. IBM SPSS Statistics software version 28 was used to analyze the data. First, The Kaiser-Meyer-Olkin (KMO) test was conducted to decide whether the data is appropriate for EFA. KMO value was found to be 0.91, which is greater than the threshold value of 0.90 [84]. Furthermore, Bartlett test result was significant ($p<.001$), showing adequate correlations [85].

The results of EFA showed that a total of nine factors with eigenvalues greater than 1 was revealed with a total variance explanation of 66.14%. The number of factors was same as the number of factors in Agile Software Project Success Model. Only three items ('ORG4', 'TEAM4', 'TEAM5'), item loadings lower than 0.60, were drawn from the items list. Furthermore, all Cronbach's alpha values were greater than the threshold value of 0.70, indicating the adequate reliability [86]. The final item list with their corresponding constructs is shown in Appendix C - Table 9.

B. CONFIRMATORY FACTOR ANALYSIS

CFA was conducted to examine reliability and validity of the 9 constructs, described by 45 items. IBM AMOS 20.0 was used for the analysis. The initial analysis of CFA revealed that model fit values, as shown in Appendix C - Table 10, achieved acceptable values. The division of x^2 value to degrees of freedom, root-mean-square of approximation (RMSEA), goodness-of- fit index (GFI), adjusted goodness-of-fit index (AGFI), normed fit index (NFI), and comparative fit index (CFI) were examined according to the recommendations of Hair et al. [87]. All the values are within acceptable levels [88], [89], [90], [91], [92], [93], [94], suggesting that the model demonstrated good model fit to the data.

Furthermore, the convergent and discriminant validity of the constructs were also examined. Convergent validity was tested through standardized factor loadings, composite reliability (CR), t-values, and average variance extracted

TABLE 5. Demographic profiles of the respondents.

Country	Turkey	UK	Germany	Other			
(%)	73.32	13.76	6.21	6.71			
Industry	Banking	Stock Exchange	Software	Marketing/Retail	Manufacturing	Aviation	Other
(%)	30.70	18.96	15.44	7.05	4.87	4.36	18.62
Job Role	Developer	Business Analyst	Scrum Master	Project Manager	Product Owner	Tester	
(%)	37.08	17.79	17.62	11.74	8.72	7.05	
Work Experience	0 – 2 years	2 – 5 years	5 – 10 years	10 -15 years	15 years or above		
(%)	12.92	32.55	38.09	14.77	1.68		
Work Model	Hybrid	On-site	Remote				
(%)	48.83	28.52	22.65				
Agile Methodologies	Scrum	Kanban	XP	Lean			
(%)	68.12	29.36	2.01	0.50			

TABLE 6. Estimation of the structural model.

	Process efficiency	**Sustainable software product quality**	**Stakeholder satisfaction**
R^2 **(%)**	0.51	0.39	0.51
Customer factors	0.27**	0.30**	0.38**
Team factors	0.27**	0.09*	0.27**
Organizational factors	0.10*	-0.02	0.10*
Agile process factors	0.36**	0.41**	0.33**
Technical factors	0.21**	0.12**	0.05
Project factors	-0.27**	-0.16**	-0.19**

*: $p < 0.05$; **: $p < 0.01$.

(AVE), as shown in Appendix C - Table 11 and Table 12. All CR values were greater than the 0.70 threshold, showing that the internal consistency in the model is adequate [95]. At the 0.95 confidence level, all the item t-values were greater than the critical value of 1.96 [96]. Furthermore, AVE values surpassed the threshold of 0.50 [97], [98]. Thus, the convergent validity was confirmed.

Discriminant validity was assessed using the Fornell-Larcker criterion in which the square root of the AVE value of each construct is required to be greater than the correlations between the construct and the remaining ones [86]. The test results for discriminant validity are shown in Appendix C - Table 12, showing that the discriminant validity was achieved.

C. STRUCTURAL MODEL

In the structural model, the effects of each agile critical success factors (customer factors, team factors, organizational factors, agile project factors, technical factors, project factors) on each agile project success measures (process efficiency, sustainable software product quality, stakeholder satisfaction) will be analyzed. First, the fits statistics of the research model was tested. The results showed that all the fit-indices, the absolute fit indices (RMSEA = 0.04, GFI = 0.88, and AGFI = 0.86), incremental fit indices (NFI = 0.88 and CFI = 0.94), and Chi-Square/degrees of freedom (at 1.85), are within the recommended values [88], [89], [90], [91], [92], [93], [94], suggesting that the model fits to the data.

According to the results, the model presented the substantial variance in process efficiency ($R^2 = 0.51$), sustainable software product quality ($R^2 = 0.39$), and stakeholder satisfaction ($R^2 = 0.51$). A total of 18 hypotheses were tested and two hypotheses were rejected. Table 6 shows the results with the path coefficients and their significance levels.

The results showed that customer factors, team factors, agile process factors, and project factors were found to be significant predictors of the agile project success in terms of process efficiency, sustainable software product quality, and stakeholder satisfaction. Agile process factor is the strongest predictor for sustainable software product quality with a coefficient of 0.41 ($p < 0.01$), compared with team, customer, organizational, technical, and project factors. This shows that the availability of strong communication among team members, applying agile-related events, agile-friendly progress tracking/controlling mechanism, and right amount of documentation enhance sustainable software product quality. Similarly, customer factor is a strong predictor for stakeholder satisfaction, sustainable software product quality, and process efficiency with a coefficient of 0.38, 0.30, 0.27 ($p < 0.01$), respectively. Developing customer relationships, enhancing customer experience, prioritizing customer support, and providing adequate training may help to accomplish project success.

The effect of technical factors on stakeholder satisfaction was found to be insignificant with a coefficient of 0.05 ($p > 0.05$). Although technical factors aim to enhance code quality and maintainability, their direct relationship with stakeholder satisfaction may be difficult to establish. Similarly, the effect of organizational factors on sustainable software product quality was found to be insignificant with a coefficient of -0.02 ($p > 0.05$). While organizational factors help establish a supportive culture and structure, the actual quality of software products may depend on technical competencies, daily operational practices rather than on overarching organizational factors.

A total of 6 constructs, comprising ''customer factors'', ''team factors'', ''organizational factors'', ''agile process factors'', ''technical factors'', and ''project factors'', explains the 51% total variance of process efficiency. Furthermore, ''customer factors'', ''team factors'', ''agile process factors'', ''technical factors'', and ''project factors'' explain the 39% total variance of sustainable software product quality. Lastly, a total of 5 constructs, comprising ''customer factors'', ''team factors'', ''organizational factors'', ''agile process factors'', and ''project factors'', explains the 51% total variance of stakeholder satisfaction.

VI. DISCUSSION OF RESULTS

The objective of this study is to determine the CSFs for the success of agile projects, and to develop an Agile Software Project Success Model to understand the relationship between CSFs and software project success measures. According to the results, agile process factors and customer factors were found to be the strongest predictor for sustainable software product quality, outperforming other factors such as team, organizational, technical, and project factors. This highlights the importance of effective communication and coordination, applying agile events, agile-oriented progress tracking and controlling mechanisms, and appropriate documentation in enhancing sustainable software product quality. Agile project factors were also found to be important predictors for process efficiency, and stakeholder satisfaction. The research of Chow and Cao [8] also supports the importance of proper documentation, progress tracking and controlling mechanisms, and daily communication in achieving agile software project success. During meetings with agile practitioners highlighted that effective communication and coordination play a crucial role in facilitating cohesive collaboration among team members. They emphasized the significance of agile events, such as daily stand-up meetings and sprint retrospectives, as essential frameworks for iterative development. These events facilitate interdisciplinary engagement and adaptive decision-making. Furthermore, agile practitioners underscored the crucial role of agile-oriented progress tracking and controlling mechanisms, such as burndown charts and Kanban boards. That enables project teams to monitor progress, identify bottlenecks, and proactively manage project dynamics for successful outcomes.

Customer factors, including customer support, good customer relationship, customer training in IT domain, and customer experience has a positive effect on agile software development project success measures. Based on the previous literature, customer support is also a significant contributor to the success of agile software development projects [8], [9], [12]. Similarly, Jun et al. [63] showed that customer training and experience play a positive role in project success. During meetings, agile practitioners emphasized a strong customer relationship throughout the project lifecycle as strong customer relationship encourages trust and effective communication. Furthermore, they underlined the importance of providing customer training to guarantee the successful adoption and use of IT solutions.

Team factors were also found to be significant predictors for process efficiency, sustainable product quality, and stakeholder satisfaction. Previous studies have also indicated that team factors play a significant role in determining the success of software projects [6], [8], [9], [10], [12], [58], [59]. Specifically, the capability, experience, and skills of team members have been shown to have a positive impact on software project success. This emphasizes the significance of having a competent and skilled team in place to ensure successful agile project outcomes.

Organizational factors such as agile style work environment, cooperative management style, organizationally accepted change management, and upper management support have relatively lower positive impact on process efficiency and stakeholder satisfaction. Furthermore, the effect of organizational factors on sustainable software product quality was found to be insignificant. This shows that although organizational factors are essential, they might not distinctly guarantee improved software product quality. Organizational characteristics offer a conducive atmosphere for agile practices and help to establish a supportive culture and structure, the actual quality of software products depends on other factors such as technical competencies, daily operational practices, and the adaptability of development teams. Similarly, Chow and Cao [8] did not find statistical evidence to support the impact of organizational culture for agile software project success. The Covid-19 pandemic has introduced different working models into our lives, and in the last four years, there has been a rapid adaptation to remote and hybrid working models. As a result, the effect of organizational factors on agile project success may have changed in recent years.

The other finding of the current study is that the technical factors have positive impacts on process efficiency and sustainable software product quality. This finding is similar to the findings of Chow and Cao. In their study, Chow and Cao [8] provides further evidence on the significance of software testing process, programming standards adherence, refactoring activities, simplicity, and regular delivery of working software in attaining agile software project success. Meetings with agile practitioners underlined the significance of software testing processes in project management, the importance of adhering to rigorous programming standards and refactoring activities to provide software quality, maintainability and minimize technical dept. However, the effect of technical factors on stakeholder satisfaction was found to be insignificant. This shows that while technical factors like system performance or code quality are crucial components of software development, their direct effect on stakeholder satisfaction may not be immediately apparent. Customers and upper management may prioritize visible product improvements. Furthermore, adopting cutting-edge technologies may not always result in enhanced customer satisfaction if the new

TABLE 7. Success factors of the agile project.

Constructs	Abbreviations	Survey questions
Organizational factors	ORGN1 ORGN2 ORGN3 ORGN4 ORGN5	The agile project team members worked in an agile-style work environment, e.g., proper working place, suitable environment for remote/onsite pair programming, usage of tools like video talk, desktop sharing, etc. [11]. There was organizationally accepted cooperative management style instead of hierarchal [11]. There was organizationally accepted change management and structure [14]. The organizational and project vision and mission were well defined [6], [14]. The project received upper management support [11].
Team factors	TEAM1 TEAM2 TEAM3 TEAM4 TEAM5 TEAM6 TEAM7	The agile project team members had technical capability [11]. The agile project team consisted of experienced team members [6], [11], [44]. The agile project team members had appropriate technical training on agile procedures and subject matter [11]. The project manager had a knowledge of agile principles and procedures [11]. The appointed project management had experience to manage the project effectively [44]. The agile project team members were highly motivated and committed to success of the project [11]. The agile project team consisted of self-organized team members, e.g., relying on team members who can work together to solve problems and adjust to changing circumstances [11].
Customer factors	CUST1 CUST2 CUST3 CUST4	The project had strong customer support throughout the project [11]. Agile project team members had a good relationship with the customer [11]. There was appropriate technical training for customers throughout the project, including training on agile procedures and subject matter [6], [11] The customer's experience in its own domain was high [6].
Technical factors	TECH1 TECH2 TECH3 TECH4 TECH5 TECH6 TECH7	The project followed an efficient software testing process for each iteration [11]. The agile project team followed a well-defined coding standard [11]. The agile project team followed refactoring activities to provide optimal results and to adapt well changes in requirements [11]. The agile project team consistently delivered working software in short periods of time [11]. The agile project team followed appropriate platforms, technologies, and tools for agile practice [11]. The project had up-front risk analysis done and assessed for the use of the agile methodology [11]. The project pursued simple design, e.g., keeping code and design simply, backlog refinement, etc. [11].
Agile process factors	AGPF1 AGPF2 AGPF3 AGPF4	The project had strong communication and coordination, e.g., instant communication channels (between agile team members, between agile team members and upper management, and between agile team members and customers), having cycle meetings, etc. [11]. The project applied agile related events properly, e.g., daily meetings, sprint planning, etc. [11]. The project was followed with an agile-friendly progress tracking/controlling mechanism, e.g., product backlog, sprint backlog, Kanban board, release & iteration burndown chart, etc. [11]. The project had the right amount of documentation for software development [11].
Project factors	PRJF1 PRJF2 PRJF3 PRJF4 PRJF5 PRJF6	The project had high level of criticality (an example of critical projects is loss of essential monies, in which the company could go bankrupt due to a defect in the program). The project had a high level of urgency [6]. The project had high level of technical complexity [14]. The project had technological uncertainty, e.g., no established procedures/practices for software development meeting the business requirements [14]. Customer requirements identified at beginning of the project quite changed during the project [14]. The number of people on team was large (The agile team typically consists of 10 or less members) [14].

features developed with these technologies do not correspond to the customer.

Another result revealed that, project factors were found to have a negative effect on process efficiency, sustainable software product quality, and stakeholder satisfaction. Agile practitioners in group and one-on-one meetings have stated concerns about the negative consequences of technological uncertainty, technical complexity, urgency, project criticality, large project size, and changes in specifications in agile software project management. Technical complexity presents obstacles in design and implementation, frequently resulting in delays and cost increases. Furthermore, the large project size may cause communication and coordination issues leading to delays and lost productivity.

The developed model is particularly useful for companies working on agile software projects to raise awareness. First, agile practitioners may benefit from using the proposed model during planning and execution of agile software

TABLE 8. Perception of success of the agile project.

Constructs	Abbreviations	Survey questions
Process efficiency	EFFC1	The project was completed on time [11], [14].
	EFFC2	The project was completed within the planned budget [11], [14].
	EFFC3	The project was completed within the agreed scope [11], [14].
Software product quality	QUAL1	The project was successful in terms of software reliability, e.g. The software run without error for a specified amount of time [14], [26].
	QUAL2	The software delivered was considered as successful in terms of security, e.g., the ability of product to protect itself from intentional or accidental intrusion [6], [27].
	QUAL3	The software was successful in terms of the maintainability, e.g., upgrades of the operating system, correcting implementation failures of the system and etc. [6], [27].
	QUAL4	The system meets users' intended functional requirements [6], [14], [27].
	QUAL5	The project was successful in terms of software performance efficiency [27].
	QUAL6	The project was successful in terms of software compatibility, e.g., the compatibility of your website with different browsers such as Google Chrome, Firefox etc. [27].
	QUAL7	The project was successful in terms of software portability (transferability), e.g., if the effort required to move the program is significantly less than that required to initially implement it, and the effort is small in absolute terms, the program is portable [27].
	QUAL8	The application developed is easy to use [14], [27].
	QUAL9	The project was successful in terms of quality [11], [27].
Stakeholder satisfaction	SATF1	Customers were satisfied with the system delivered [14].
	SATF2	The project team was satisfied with the system delivered [14].
	SATF3	Top level management was satisfied with the system delivered [14].

projects. Second, agile methodologies draw attention to customer participation and feedback. Organizations may establish strong communication channels among stakeholders, set clear expectations, and control requirements through priority and backlog management strategies. Third, according to the Agile Manifesto, "working software over comprehensive documentation" [46] enhances efficiency. On the other hand, having a well-documented project history may be difficult during knowledge transfer and maintenance. To reduce this risk, organizations may use the right amount of documentation. Critical project artifacts may be developed to balance the necessity for documentation. Fourth, enhancing the professional capabilities of team members and customers through training is critical in managing risks and resolving the limits of agile techniques. Team members, stakeholders, and upper management may understand agile concepts and practices through training. Therefore, organizations should encourage employees to attend training, workshops, and conferences. Fifth, the sixth principle of the Agile Manifesto emphasizes the importance of face-to-face conversation for effective information conveying within a development team. However, the working style of companies has recently changed because of the COVID-19 pandemic. The "agile-style work environment" factor emphasizes that information can be conveyed efficiently and effectively using the right methods and tools in a remote work environment.

VII. RESEARCH LIMITATIONS AND FUTURE STUDIES

This study encompasses several limitations that must be acknowledged. First, all agile practitioners that attended one-on-one and group meetings work in companies from Turkey. Different country profiles of the agile practitioners may contribute to the current study. Second, the results of the survey data do not comprise all agile methodologies such as Lean, XP. Only 15 survey participants used Lean and XP methodologies in their projects reported. Because most of participants used Scrum and Kanban, there is a bias toward Scrum and Kanban in the data reported. Third limitation is about sectoral differences in which most of the respondents come from finance related industry and answered the questionnaire according to financial-related agile projects done.

As a further study, more data may be collected from agile practitioners working with different type of agile project management approach to have a more comprehensive understanding about the effects of CSFs on agile project success measures. Similarly, to understand the sector differences such as financial, marketing, or other sectors, more data may be gathered to see sectoral differences in agile project management approach. Second, a detailed study may be conducted to enhance literature about sustainability in agile project management and software product quality. Third, the explained variances for process efficiency, sustainable software product quality, and stakeholder satisfaction were 51%, 39%, and 51%, respectively. Although these findings are encouraging, additional factors may be examined to reveal their effects on project success measures. Fourth, risk and failure factors were incorporated into the current study. However, more detailed study may be conducted to enhance the literature about potential challenges, such as scalability issues or difficulties in maintaining agility as the project scales, are meticulously addressed through strategic planning and continuous refinement of agile practices. Fifth, qualitative approaches, such as in-depth interviews and case studies, may be used in the upcoming studies to completely capture the depth of insights that may not be entirely captured by quantitative analysis alone.

VIII. CONCLUSION

In this study, the Agile Software Project Success Model, derived from a systematic literature review for last 24 years, agile practitioners' input through one-on-one and group meetings, and empirical analysis, identifies critical success factors and complex relationships specific to agile software development and management. Agile critical success factors were defined with the ''Customer factors'', ''Team factors'', ''Organizational factors'', ''Agile Process factors'', ''Technical factors'' and ''Project factors''. Furthermore, agile project success measures were defined with the ''Process efficiency'', ''Sustainable software product quality'', and ''Stakeholder satisfaction''. In contrast to previous studies, this study emphasizes empirical validation and practical applicability, providing a valuable tool for researchers and practitioners. The study presents an achievement in the field, combining theoretical insights with practical knowledge and filling a substantial gap in agile project success management. It highlights the significance of understanding the factors in the success of agile projects. The current study is a useful guide for researchers in academia and agile practitioners for the success of agile projects.

APPENDIX A
SELECTED ARTICLES
REFERENCES

SLR1 Abrar, M. F.; Khan, M. S.; Ali, S.; Ali, U.; Majeed, M. F.; Ali, A.; Amin, B.; Rasheed, N. Motivators for Large-Scale Agile Adoption from Management Perspective: A Systematic Literature Review. IEEE Access 2019, 7, 22660–22674. https://doi.org/10.1109/ACCESS.2019.2896212.

SLR2 Ahimbisibwe, A.; Cavana, R. Y.; Daellenbach, U. A Contingency Fit Model of Critical Success Factors for Software Development Projects: A Comparison of Agile and Traditional Plan-Based Methodologies. J. Enterp. Inf. Manag. 2015, 28 (1), 7–33. https://doi.org/10.1108/JEIM-08-2013-0060.

SLR3 Ahimbisibwe, A.; Daellenbach, U.; Cavana, R. Y. Empirical Comparison of Traditional Plan-Based and Agile Methodologies: Critical Success Factors for Outsourced Software Development Projects from Vendors' Perspective. J. Enterp. Inf. Manag. 2017, 30 (3), 400–453. https://doi.org/10.1108/JEIM-06-2015-0056.

SLR4 Anthony Jnr, B. Validating the Usability Attributes of AHP-Software Risk Prioritization Model Using Partial Least Square-Structural Equation Modeling. J. Sci. Technol. Policy Manag. 2019, 10 (2), 404–430. https://doi.org/10.1108/JSTPM-06-2018-0060.

SLR5 Arcos-Medina, G.; Mauricio, D. Aspects of Software Quality Applied to the Process of Agile Software Development: A Systematic Literature Review. Int. J. Syst. Assur. Eng. Manag. 2019, 10 (5), 867–897. https://doi.org/10.1007/s13198-019-00840-7.

SLR6 Chow, T.; Cao, D.-B. A Survey Study of Critical Success Factors in Agile Software Projects. J. Syst. Softw. 2008, 81 (6), 961–971. https://doi.org/10.1016/j.jss.2007.08.020.

SLR7 Engelbrecht, J.; Johnston, K. A.; Hooper, V. The Influence of Business Managers' IT Competence on IT Project Success. Int. J. Proj. Manag. 2017, 35 (6), 994–1005.

SLR8 Fatema, I.; Sakib, K. Factors Influencing Productivity of Agile Software Development Teamwork: A Qualitative System Dynamics Approach. In 2017 24th Asia-Pacific Software Engineering Conference (APSEC), 2017, 737–742. https://doi.org/10.1109/APSEC.2017.95.

SLR9 Fortune, J.; White, D. Framing of Project Critical Success Factors by a Systems Model. Int. J. Proj. Manag. 2006, 24, 53–65. https://doi.org/10.1016/j.ijproman.2005.07.004.

SLR10 Garousi, V.; Tarhan, A.; Pfahl, D.; Coşkunçay, A.; Demirörs, O. Correlation of Critical Success Factors with Success of Software Projects: An Empirical Investigation. Softw. Qual. J. 2019, 27, 429–493. https://doi.org/10.1007/s11219-018-9419-5.

SLR11 Howell, D.; Windahl, C.; Seidel, R. A Project Contingency Framework Based on Uncertainty and Its Consequences. Int. J. Proj. Manag. 2010, 28 (3), 256–264. https://doi.org/10.1016/j.ijproman.2009.06.002.

SLR12 Hughes, D. L.; Rana, N. P.; Dwivedi, Y. K. Elucidation of IS Project Success Factors: An Interpretive Structural Modelling Approach. Ann. Oper. Res. 2019, 285 (1–2), 35–66. https://doi.org/10.1007/s10479-019-03146-w.

SLR13 Inayat, I.; Salim, S. S.; Marczak, S.; Daneva, M.; Shamshirband, S. A Systematic Literature Review on Agile Requirements Engineering Practices and Challenges. Comput. Human Behav. 2015, 51, 915–929. https://doi.org/10.1016/j.chb.2014.10.046.

SLR14 Joslin, R.; Müller, R. Relationships between a Project Management Methodology and Project Success in Different Project Governance Contexts. Int. J. Proj. Manag. 2015, 33 (6), 1377–1392. https://doi.org/10.1016/j.ijproman.2015.03.005.

SLR15 Joslin, R.; Müller, R. The Relationship between Project Governance and Project Success. Int. J. Proj. Manag. 2016, 34 (4), 613–626. https://doi.org/10.1016/j.ijproman.2016.01.008.

SLR16 Kanjanda, T.; Tuan, N.-T. A Systemic Exploration of the Risk Factors in Zimbabwean Information Technology Projects. Syst. Pract. action Res. 2020, 33 (1), 77–93. https://doi.org/10.1007/s11213-019-09515-7. 17

SLR17 Lehtinen, T. O. A.; Mäntylä, M. V; Vanhanen, J.; Itkonen, J.; Lassenius, C. Perceived Causes of Software Project Failures – An Analysis of Their Relationships. Inf. Softw. Technol. 2014, 56 (6), 623–643. https://doi.org/10.1016/j.infsof.2014.01.015.

SLR18 Menezes, J.; Gusmão, C.; Moura, H. Risk Factors in Software Development Projects: A Systematic Literature Review. Softw. Qual. J. 2019, 27 (3), 1149–1174. https://doi.org/10.1007/s11219-018-9427-5.

SLR19 Misra, S. C.; Kumar, V.; Kumar, U. Identifying Some Important Success Factors in Adopting Agile Software Development Practices. J. Syst. Softw. 2009, 82 (11), 1869–1890. https://doi.org/10.1016/j.jss.2009.05.052.

SLR20 Perera, C.; Perera, I. The Impact of Client Involvement towards Agile Project Success in Sri Lankan Software Industry. MERCon 2019 - Proceedings, 5th Int. Multidiscip. Moratuwa Eng. Res. Conf. 2019, 279–284. https://doi.org/10.1109/MERCon.2019.8818800.

SLR21 Perkusich, M.; Gorgônio, K. C.; Almeida, H.; Perkusich, A. Assisting the Continuous Improvement of Scrum Projects Using Metrics and Bayesian Networks. J. Softw. Evol. Process 2017, 29 (6), 1–17. https://doi.org/10.1002/smr.1835.

SLR22 Rashid, N.; Khan, S. U. Using Agile Methods for the Development of Green and Sustainable Software: Success Factors for GSD Vendors. J. Softw. Evol. Process 2018, 30 (8), 1–29. https://doi.org/10.1002/smr.1927.

SLR23 Rashid, N.; Khan, S. U.; Khan, H. U.; Ilyas, M. Green-Agile Maturity Model: An Evaluation Framework for Global Software Development Vendors. IEEE Access 2021, 9, 71868–71886. https://doi.org/10.1109/ACCESS.2021.3079194

SLR24 Serrador, P.; Pinto, J. K. Does Agile Work? - A Quantitative Analysis of Agile Project Success. Int. J. Proj. Manag. 2015, 33 (5), 1040–1051. https://doi.org/10.1016/j.ijproman.2015.01.006.

SLR25 Sheffield, J.; Lemétayer, J. Factors Associated with the Software Development Agility of Successful Projects. Int. J. Proj. Manag. 2013, 31 (3), 459–472. https://doi.org/10.1016/j.ijproman.2012.09.011.

SLR26 Shrivastava, S. V.; Rathod, U. Categorization of Risk Factors for Distributed Agile Projects. Inf. Softw. Technol. 2015, 58, 373–387. https://doi.org/10.1016/j.infsof.2014.07.007.

SLR27 Shrivastava, S. V.; Rathod, U. A Risk Management Framework for Distributed Agile Projects. Inf. Softw. Technol. 2017, 85, 1–15. https://doi.org/10.1016/j.infsof.2016.12.005.

SLR28 Silva, K. M. B. Da; Santos, S. C. Dos. Critical Factors in Agile Software Projects According to People, Process and Technology Perspective. Proc. - 6th Brazilian Work. Agil. Methods, WBMA 2015 2015, 48–54. https://doi.org/10.1109/WBMA.2015.19.

SLR29 Stankovic, D.; Nikolic, V.; Djordjevic, M.; Cao, D.-B. A Survey Study of Critical Success Factors in Agile Software Projects in Former Yugoslavia IT Companies. J. Syst. Softw. 2013, 86 (6), 1663–1678. https://doi.org/10.1016/j.jss.2013.02.027.

SLR30 Purna Sudhakar, G. A Model of Critical Success Factors for Software Projects. J. Enterp. Inf. Manag. 2012, 25 (6), 537–558. https://doi.org/10.1108/17410391211272829.

SLR31 Tam, C.; Moura, E. J. da C.; Oliveira, T.; Varajão, J. The Factors Influencing the Success of On-Going Agile Software Development Projects. Int. J. Proj. Manag. 2020, 38 (3), 165–176. https://doi.org/10.1016/j.ijproman.2020.02.001.

SLR32 Wan, J.; Wang, R. Empirical Research on Critical Success Factors of Agile Software Process Improvement. J. Softw. Eng. Appl. 2010, 03, 1131–1140. https://doi.org/10.4236/jsea.2010.312132.

SLR33 Yeo, K. T. Critical Failure Factors in Information System Projects. Int. J. Proj. Manag. 2002, 20 (3), 241–246. https://doi.org/10.1016/S0263-7863(01)00075-8.

APPENDIX B
SURVEY QUESTIONS

See table 7 and 8.

APPENDIX C

See table 9–12.

TABLE 9. Standardized item loadings, variance explained, and cronbach's alpha values.

Constructs	Items	Standardized Item Loadings	Variance Explained (%)	Cronbach α	Constructs	Items	Standardized Item Loadings	Variance Explained (%)	Cronbach α	Constructs	Items	Standardized Item Loadings	Variance Explained (%)	Cronbach α
Customer factors	CUST1 CUST2 CUST3 CUST4	0.78 0.77 0.78 0.80	6.08	0.83	Agile process factors	AGPF1 AGPF2 AGPF3 AGPF4	0.74 0.77 0.76 0.78	5.93	0.82	Process efficiency	EFFC1 EFFC2 EFFC3	0.77 0.76 0.75	4.76	0.84
Team factors	TEAM1 TEAM2 TEAM3 TEAM6 TEAM7	0.82 0.77 0.77 0.73 0.74	7.01	0.85	Organizational factor	ORGN1 ORGN2 ORGN3 ORGN5	0.79 0.76 0.81 0.80	6.04	0.82	Stakeholder satisfaction	SATF1 SATF2 SATF3	0.73 0.79 0.74	4.73	0.83
Technical factors	TECH1 TECH2 TECH3 TECH4 TECH5 TECH6 TECH7	0.73 0.75 0.73 0.77 0.78 0.79 0.76	9.18	0.88	Project factors	PRJF1 PRJF2 PRJF3 PRJF4 PRJF5 PRJF6	0.75 0.79 0.75 0.76 0.74 0.76	8.57	0.87	Sustainable software product quality	QUAL1 QUAL2 QUAL3 QUAL4 QUAL5 QUAL6 QUAL7 QUAL8 QUAL9	0.78 0.78 0.75 0.81 0.78 0.79 0.78 0.80 0.75	13.83	0.93

TABLE 10. The model fit statistics of the confirmatory factor analysis.

Fit index	Recommended value	Observed value	Model fitting	References
x^2 / df	≤ 3	1.84	Conform	[93]
RMSEA	≤ 0.08	0.04	Conform	[89]
GFI	≥ 0.80	0.88	Conform	[91]
AGFI	≥ 0.80	0.86	Conform	[88], [91], [92]
CFI	≥ 0.90	0.94	Conform	[87]
NFI	≥ 0.80	0.88	Conform	[90]

TABLE 11. Convergent validity results.

Constructs	Items	CFA Loadings	t-value	Constructs	Items	CFA Loadings	t-value	Constructs	Items	CFA Loadings	t-value
Customer factors	CUST1 CUST2 CUST3 CUST4	0.76 0.78 0.75 0.67	17.07 17.55 17.07 15.24	Agile process factors	AGPF1 AGPF2 AGPF3 AGPF4	0.76 0.74 0.72 0.72	16.16 16.57 16.28 16.16	Process efficiency	EFFC1 EFFC2 EFFC3	0.79 0.79 0.81	19.26 19.32 19.32
Team factors	TEAM1 TEAM2 TEAM3 TEAM6 TEAM7	0.64 0.64 0.75 0.79 0.79	15.35 15.25 17.98 15.35 18.97	Organizational factor	ORGN1 ORGN2 ORGN3 ORGN5	0.75 0.71 0.73 0.74	15.81 15.06 15.06 15.70	Stakeholder satisfaction	SATF1 SATF2 SATF3	0.81 0.74 0.80	17.95 17.95 17.78
Technical factors	TECH1 TECH2 TECH3 TECH4 TECH5 TECH6 TECH7	0.69 0.69 0.67 0.74 0.75 0.77 0.71	15.22 15.22 14.79 16.04 16.27 16.67 15.50	Project factors	PRJF1 PRJF2 PRJF3 PRJF4 PRJF5 PRJF6	0.74 0.74 0.71 0.72 0.69 0.74	16.99 16.99 16.46 16.52 15.85 16.99	Sustainable software product quality	QUAL1 QUAL2 QUAL3 QUAL4 QUAL5 QUAL6 QUAL7 QUAL8 QUAL9	0.77 0.79 0.77 0.79 0.78 0.80 0.79 0.80 0.74	20.33 20.84 20.27 20.86 19.42 21.17 21.09 21.29 19.42

TABLE 12. Discriminant validity results.

	AVE	CR	Sustainable software product quality	Customer factors	Team factors	Organizational factors	Agile process factors	Technical factors	Project factors	Process efficiency	Stakeholder satisfaction
Sustainable software product quality	0.61	0.93	**0.78**								
Customer factors	0.55	0.83	0.38	**0.74**							
Team factors	0.52	0.84	0.25	0.23	**0.72**						
Organizational factors	0.54	0.82	-0.11	0.04	-0.08	**0.73**					
Agile process factors	0.54	0.82	0.48	0.19	0.22	-0.13	**0.73**				
Technical factors	0.51	0.88	0.1	-0.03	-0.02	0.04	-0.06	**0.72**			
Project factors	0.52	0.87	-0.22	-0.01	-0.09	0.3	-0.11	-0.08	**0.72**		
Process efficiency	0.63	0.84	0.46	0.39	0.41	-0.03	0.46	0.2	-0.32	**0.79**	
Stakeholder satisfaction	0.61	0.83	0.51	0.5	0.43	0.01	0.45	0.03	-0.21	0.53	**0.78**

Diagonal values indicate the square roots of AVEs.

ACKNOWLEDGMENT

The authors express gratitude toward the agile practitioners who took part in the survey, as well as to the six agile practitioners who attended individual and group meetings.

REFERENCES

[1] *Agile 101, The Agile Alliance*. Accessed: Jun. 22, 2023. [Online]. Available: https://www.agilealliance.org/agile101/

[2] F. K. Y. Chan and J. Y. L. Thong, "Acceptance of agile methodologies: A critical review and conceptual framework," *Decis. Support Syst.*, vol. 46, no. 4, pp. 803–814, Mar. 2009, doi: 10.1016/j.dss.2008.11.009.

[3] J. A. Klünder, P. Hohl, N. Prenner, and K. Schneider, "Transformation towards agile software product line engineering in large companies: A literature review," *J. Softw., Evol. Process*, vol. 31, no. 5, p. e2168, May 2019, doi: 10.1002/smr.2168.

[4] J. Johnson, *CHAOS 2020: Beyond Infinity Overview*. Boston, MA, USA: The Standish Group International, 2021.

[5] D. Milosevic and P. Patanakul, "Standardized project management may increase development projects success," *Int. J. Project Manag.*, vol. 23, no. 3, pp. 181–192, Apr. 2005, doi: 10.1016/j.ijproman.2004.11.002.

[6] V. Garousi, A. Tarhan, D. Pfahl, A. Coşkunçay, and O. Demirörs, "Correlation of critical success factors with success of software projects: An empirical investigation," *Softw. Quality J.*, vol. 27, no. 1, pp. 429–493, Mar. 2019, doi: 10.1007/s11219-018-9419-5.

[7] H. N. N. Mohd and S. Shamsul, "Critical success factors for software projects: A comparative study," *Sci. Res. Essays*, vol. 6, no. 10, pp. 2174–2186, May 2011, doi: 10.5897/sre10.1171.

[8] T. Chow and D. B. Cao, "A survey study of critical success factors in agile software projects," *J. Syst. Softw.*, vol. 81, no. 6, pp. 961–971, 2008, doi: 10.1016/j.jss.2007.08.020.

[9] S. C. Misra, V. Kumar, and U. Kumar, "Identifying some important success factors in adopting agile software development practices," *J. Syst. Softw.*, vol. 82, no. 11, pp. 1869–1890, Nov. 2009, doi: 10.1016/j.jss.2009.05.052.

[10] J. Wan and R. Wang, "Empirical research on critical success factors of agile software process improvement," *J. Softw. Eng. Appl.*, vol. 3, no. 12, pp. 1131–1140, 2010, doi: 10.4236/jsea.2010.312132.

[11] D. Stankovic, V. Nikolic, M. Djordjevic, and D.-B. Cao, "A survey study of critical success factors in agile software projects in former yugoslavia IT companies," *J. Syst. Softw.*, vol. 86, no. 6, pp. 1663–1678, Jun. 2013, doi: 10.1016/j.jss.2013.02.027.

[12] A. Ahimbisibwe, R. Y. Cavana, and U. Daellenbach, "A contingency fit model of critical success factors for software development projects: A comparison of agile and traditional plan-based methodologies," *J. Enterprise Inf. Manag.*, vol. 28, no. 1, pp. 7–33, Feb. 2015, doi: 10.1108/jeim-08-2013-0060.

[13] K. M. B. D. Silva and S. C. D. Santos, "Critical factors in agile software projects according to people, process and technology perspective," in *Proc. 6th Brazilian Workshop Agile Methods (WBMA)*, Oct. 2015, pp. 48–54, doi: 10.1109/WBMA.2015.19.

[14] A. Ahimbisibwe, U. Daellenbach, and R. Y. Cavana, "Empirical comparison of traditional plan-based and agile methodologies: Critical success factors for outsourced software development projects from vendors' perspective," *J. Enterprise Inf. Manag.*, vol. 30, no. 3, pp. 400–453, Apr. 2017, doi: 10.1108/jeim-06-2015-0056.

[15] 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, pp. 1–29, Aug. 2018, doi: 10.1002/smr.1927.

[16] G. Arcos-Medina and D. Mauricio, "Aspects of software quality applied to the process of agile software development: A systematic literature review," *Int. J. Syst. Assurance Eng. Manag.*, vol. 10, no. 5, pp. 867–897, Oct. 2019, doi: 10.1007/s13198-019-00840-7.

[17] S. V. Shrivastava and U. Rathod, "Categorization of risk factors for distributed agile projects," *Inf. Softw. Technol.*, vol. 58, pp. 373–387, Feb. 2015, doi: 10.1016/j.infsof.2014.07.007.

[18] S. V. Shrivastava and U. Rathod, "A risk management framework for distributed agile projects," *Inf. Softw. Technol.*, vol. 85, pp. 1–15, May 2017, doi: 10.1016/j.infsof.2016.12.005.

[19] C. Perera and I. Perera, "The impact of client involvement towards agile project success in sri Lankan software industry," in *Proc. Moratuwa Eng. Res. Conf. (MERCon)*, Jul. 2019, pp. 279–284, doi: 10.1109/MERCon.2019.8818800.

[20] C. Tam, E. J. D. C. Moura, T. Oliveira, and J. Varajão, "The factors influencing the success of on-going agile software development projects," *Int. J. Project Manag.*, vol. 38, no. 3, pp. 165–176, Apr. 2020, doi: 10.1016/j.ijproman.2020.02.001.

[21] J. D. Reeves, T. Eveleigh, T. H. Holzer, and S. Sarkani, "Risk identification biases and their impact to space system development project performance," *Eng. Manag. J.*, vol. 25, no. 2, pp. 3–12, Jun. 2013, doi: 10.1080/10429247.2013.11431970.

[22] J. Menezes, C. Gusmão, and H. Moura, "Risk factors in software development projects: A systematic literature review," *Softw. Quality J.*, vol. 27, no. 3, pp. 1149–1174, Sep. 2019, doi: 10.1007/s11219-018-9427-5.

[23] B. Anthony Jnr., "Validating the usability attributes of AHP-software risk prioritization model using partial least square-structural equation modeling," *J. Sci. Technol. Policy Manag.*, vol. 10, no. 2, pp. 404–430, Jun. 2019, doi: 10.1108/jstpm-06-2018-0060.

[24] T. Kanjanda and N.-T. Tuan, "A systemic exploration of the risk factors in zimbabwean information technology projects," *Systemic Pract. Action Res.*, vol. 33, no. 1, pp. 77–93, Feb. 2020, doi: 10.1007/s11213-019-09515-7.

[25] B. Kitchenham and C. Charters, "Guidelines for performing systematic literature reviews in software engineering," Dept. Comput. Sci., Keele Univ. Durham Univ., Keele, U.K., Joint Rep. EBSE 2007-001, 2007.

[26] *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, 5th ed., Tokyo, Japan: Project Management Institute (PMI), 2013.

[27] International Organization for Standardization (ISO). *Systems and Software Engineering-Systems and Software Quality Requirements and Evaluation (Square)-System and Software Quality Models*, Standard ISO/IEC 25010, ISO, Geneva, Switzerland, 2011.

[28] D. Ciric Lalic, B. Lalic, M. Delić, D. Gracanin, and D. Stefanovic, "How project management approach impact project success? From traditional to agile," *Int. J. Manag. Projects Bus.*, vol. 15, no. 3, pp. 494–521, Apr. 2022, doi: 10.1108/ijmpb-04-2021-0108.

[29] J. Engelbrecht, K. A. Johnston, and V. Hooper, "The influence of business managers' IT competence on IT project success," *Int. J. Project Manag.*, vol. 35, no. 6, pp. 994–1005, Aug. 2017, doi: 10.1016/j.ijproman.2017.04.016.

[30] R. Joslin and R. Müller, "Relationships between a project management methodology and project success in different project governance contexts," *Int. J. Project Manag.*, vol. 33, no. 6, pp. 1377–1392, Aug. 2015, doi: 10.1016/j.ijproman.2015.03.005.

[31] R. Joslin and R. Müller, "The relationship between project governance and project success," *Int. J. Project Manag.*, vol. 34, no. 4, pp. 613–626, May 2016, doi: 10.1016/j.ijproman.2016.01.008.

[32] P. Serrador and J. K. Pinto, "Does agile work—A quantitative analysis of agile project success," *Int. J. Project Manag.*, vol. 33, no. 5, pp. 1040–1051, Jul. 2015, doi: 10.1016/j.ijproman.2015.01.006.

[33] J. Sheffield and J. Lemétayer, "Factors associated with the software development agility of successful projects," *Int. J. Project Manag.*, vol. 31, no. 3, pp. 459–472, Apr. 2013, doi: 10.1016/j.ijproman.2012.09.011.

[34] I. Inayat, S. S. Salim, S. Marczak, M. Daneva, and S. Shamshirband, "A systematic literature review on agile requirements engineering practices and challenges," *Comput. Hum. Behav.*, vol. 51, pp. 915–929, Oct. 2015, doi: 10.1016/j.chb.2014.10.046.

[35] M. Perkusich, K. C. Gorgǒnio, H. Almeida, and A. Perkusich, "Assisting the continuous improvement of scrum projects using metrics and Bayesian networks," *J. Softw., Evol. Process*, vol. 29, no. 6, pp. 1–17, Jun. 2017, doi: 10.1002/smr.1835.

[36] *CHAOS Report 2015*. Accessed: Jul. 7, 2023. [Online]. Available: https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf

[37] A. Khalifeh, P. Farrell, M. Alrousan, S. Alwardat, and M. Faisal, "Incorporating sustainability into software projects: A conceptual framework," *Int. J. Manag. Projects Bus.*, vol. 13, no. 6, pp. 1339–1361, Jul. 2020, doi: 10.1108/ijmpb-12-2019-0289.

[38] N. Condori-Fernandez and P. Lago, "Characterizing the contribution of quality requirements to software sustainability," *J. Syst. Softw.*, vol. 137, pp. 289–305, Mar. 2018, doi: 10.1016/j.jss.2017.12.005.

[39] N. Condori-Fernandez, P. Lago, M. R. Luaces, and Á. S. Places, "An action research for improving the sustainability assessment framework instruments," *Sustainability*, vol. 12, no. 4, p. 1682, Feb. 2020, doi: 10.3390/su12041682.

[40] D. F. Rico, H. H. Sayani, and R. F. Field, "History of computers, electronic commerce and agile methods," *Adv. Comput.*, vol. 73, pp. 1–55, Jan. 2008, doi: 10.1016/S0065-2458(08)00401-4.

[41] N. Chapin, J. E. Hale, K. M. Khan, J. F. Ramil, and W. Tan, "Types of software evolution and software maintenance," *J. Softw. Maintenance Evol., Res. Pract.*, vol. 13, no. 1, pp. 3–30, Jan. 2001, doi: 10.1002/smr.220.

[42] J. D. Mooney, *Developing Portable Software BT—Information Technology; Reis, R.* Boston, MA, USA: Springer, 2004, pp. 55–84.

[43] *Standard Glossary of Terms Used in Software Testing, International Software Testing Qualifications Board (ISTQB), Version 3.5*. Accessed: Jun. 7, 2022. [Online]. Available: https://www.istqb.org/

[44] D. L. Hughes, N. P. Rana, and Y. K. Dwivedi, "Elucidation of IS project success factors: An interpretive structural modelling approach," *Ann. Oper. Res.*, vol. 285, nos. 1–2, pp. 35–66, Feb. 2020, doi: 10.1007/s10479-019-03146-w.

[45] K. E. Papke-Shields and K. M. Boyer-Wright, "Strategic planning characteristics applied to project management," *Int. J. Project Manag.*, vol. 35, no. 2, pp. 169–179, Feb. 2017, doi: 10.1016/j.ijproman.2016.10.015.

[46] *Manifesto for Agile Software Development*. Accessed: Jan. 10, 2023. [Online]. Available: http://agilemanifesto.org/

[47] (2023). *Principles Behind the Agile Manifesto*. Accessed: Jan. 10, 2023. [Online]. Available: http://agilemanifesto.org/principles.html

[48] J. Fortune and D. White, "Framing of project critical success factors by a systems model," *Int. J. Project Manag.*, vol. 24, no. 1, pp. 53–65, Jan. 2006, doi: 10.1016/j.ijproman.2005.07.004.

[49] M. F. Abrar, M. S. Khan, S. Ali, U. Ali, M. F. Majeed, A. Ali, B. Amin, and N. Rasheed, "Motivators for large-scale agile adoption from management perspective: A systematic literature review," *IEEE Access*, vol. 7, pp. 22660–22674, 2019, doi: 10.1109/ACCESS.2019.2896212.

[50] T. O. A. Lehtinen, M. V. Mäntylä, J. Vanhanen, J. Itkonen, and C. Lassenius, "Perceived causes of software project failures—An analysis of their relationships," *Inf. Softw. Technol.*, vol. 56, no. 6, pp. 623–643, Jun. 2014, doi: 10.1016/j.infsof.2014.01.015.

[51] I. Fatema and K. Sakib, "Factors influencing productivity of agile software development teamwork: A qualitative system dynamics approach," in *Proc. 24th Asia–Pacific Softw. Eng. Conf. (APSEC)*, Dec. 2017, pp. 737–742, doi: 10.1109/APSEC.2017.95.

[52] N. Rashid, S. U. Khan, H. U. Khan, and M. Ilyas, "Green-agile maturity model: An evaluation framework for global software development vendors," *IEEE Access*, vol. 9, pp. 71868–71886, 2021, doi: 10.1109/ACCESS.2021.3079194.

[53] A.-A. Cucolaş and D. Russo, "The impact of working from home on the success of scrum projects: A multi-method study," *J. Syst. Softw.*, vol. 197, Mar. 2023, Art. no. 111562, doi: 10.1016/j.jss.2022.111562.

[54] G. P. Sudhakar, "A model of critical success factors for software projects," *J. Enterprise Inf. Manag.*, vol. 25, no. 6, pp. 537–558, Oct. 2012, doi: 10.1108/17410391211272829.

[55] D. Howell, C. Windahl, and R. Seidel, "A project contingency framework based on uncertainty and its consequences," *Int. J. Project Manag.*, vol. 28, no. 3, pp. 256–264, Apr. 2010, doi: 10.1016/j.ijproman.2009.06.002.

[56] K. T. Yeo, "Critical failure factors in information system projects," *Int. J. Project Manag.*, vol. 20, no. 3, pp. 241–246, Apr. 2002, doi: 10.1016/s0263-7863(01)00075-8.

[57] P. Yetton, A. Martin, R. Sharma, and K. Johnston, "A model of information systems development project performance," *Inf. Syst. J.*, vol. 10, no. 4, pp. 263–289, Oct. 2000, doi: 10.1046/j.1365-2575.2000.00088.x.

[58] 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*, D. Wells and L., Williams, Eds., Berlin, Germany: Springer, 2002, pp. 197–207.

[59] B. Boehm and R. Turner, "Using risk to balance agile and plan-driven methods," *Computer*, vol. 36, no. 6, pp. 57–66, 2003, doi: 10.1109/MC.2003.1204376.

[60] N. B. Moe, T. Dingsøyr, and T. Dybå, "A teamwork model for understanding an agile team: A case study of a scrum project," *Inf. Softw. Technol.*, vol. 52, no. 5, pp. 480–491, May 2010, doi: 10.1016/j.infsof.2009.11.004.

[61] W. F. Lemon, J. Liebowitz, J. Burn, and R. Hackney, "Information systems project failure: A comparative study of two countries," *J. Global Inf. Manag.*, vol. 10, no. 2, pp. 28–39, Apr. 2002, doi: 10.4018/jgim.2002040103.

[62] K. E. Emam and A. G. Koru, "A replicated survey of IT software project failures," *IEEE Softw.*, vol. 25, no. 5, pp. 84–90, Sep. 2008, doi: 10.1109/ms.2008.107.

[63] L. Jun, W. Qiuzhen, and M. Qingguo, "The effects of project uncertainty and risk management on IS development project performance: A vendor perspective," *Int. J. Project Manag.*, vol. 29, no. 7, pp. 923–933, Oct. 2011, doi: 10.1016/j.ijproman.2010.11.002.

[64] M. Taromirad and R. Ramsin, "An appraisal of existing evaluation frameworks for agile methodologies," in *Proc. 15th Annu. IEEE Int. Conf. Workshop Eng. Comput. Based Syst.*, Apr. 2008, pp. 418–427, doi: 10.1109/ECBS.2008.32.

[65] *The 2020 Scrum Guide*. Accessed: Dec. 11, 2022. [Online]. Available: https://scrumguides.org/scrum-guide.html

[66] V. Teslyuk, A. Batyuk, and V. Voityshyn, "Method of software development project duration estimation for scrum teams with differentiated specializations," *Systems*, vol. 10, no. 4, p. 123, Aug. 2022, doi: 10.3390/systems10040123.

[67] C. Mann and F. Maurer, "A case study on the impact of scrum on overtime and customer satisfaction," in *Proc. Agile Develop. Conf.*, Los Alamitos, CA, USA, 2005, pp. 70–79, doi: 10.1109/ADC.2005.1.

[68] M. Cohn and D. Ford, "Introducing an agile process to an organization [software development]," *Computer*, vol. 36, no. 6, pp. 74–78, 2003, doi: 10.1109/MC.2003.1204378.

[69] O. Erdogan, M. E. Pekkaya, and H. Gök, "More effective sprint retrospective with statistical analysis," *J. Softw. Evol. Process.*, vol. 30, no. 5, pp. 1–9, 2018, doi: 10.1002/smr.1933.

[70] L. Crispin and J. Gregory, *Agile Testing: A Practical Guide for Testers and Agile Teams*, 1st ed., Reading, MA, USA: Addison-Wesley, 2008, pp. 97–108.

[71] M. Fowler and K. Beck, *Refactoring: Improving the Design of Existing Code*. Reading, MA, USA: Addison-Wesley, 1999.

[72] D. Leffingwell, *Scaling Software Agility: Best Practices for Large Enterprises*. Upper Saddle River, NJ, USA: Addison-Wesley, 2008.

[73] U. Ahmad, Y. Ibrahim, and M. S. Minai, "Malaysian public–private partnerships: Risk management in build, lease, maintain and transfer projects," *Cogent Bus. Manag.*, vol. 5, no. 1, Jan. 2018, Art. no. 1550147, doi: 10.1080/23311975.2018.1550147.

[74] *Risk Management-Principles and Guidelines*, Standard ISO 31000, ISO, Geneva, Switzerland, 2018.

[75] B. G. Tavares, C. E. S. da Silva, and A. D. de Souza, "Risk management analysis in scrum software projects," *Int. Trans. Oper. Res.*, vol. 26, no. 5, pp. 1884–1905, Sep. 2019, doi: 10.1111/itor.12401.

[76] S.-J. Huang and W.-M. Han, "Exploring the relationship between software project duration and risk exposure: A cluster analysis," *Inf. Manag.*, vol. 45, no. 3, pp. 175–182, Apr. 2008, doi: 10.1016/j.im.2008.02.001.

[77] K. Beck, "Embracing change with extreme programming," *Computer*, vol. 32, no. 10, pp. 70–77, 1999, doi: 10.1109/2.796139.

[78] R. Schmidt, K. Lyytinen, M. Keil, and P. Cule, "Identifying software project risks: An international delphi study," *J. Manag. Inf. Syst.*, vol. 17, no. 4, pp. 5–36, Mar. 2001, doi: 10.1080/07421222.2001.11045662.

[79] M. I. Lunesu, R. Tonelli, L. Marchesi, and M. Marchesi, "Assessing the risk of software development in agile methodologies using simulation," *IEEE Access*, vol. 9, pp. 134240–134258, 2021, doi: 10.1109/ACCESS.2021.3115941.

[80] A. S. Koch, *Agile Software Development: Evaluating the Methods for Your Organization*. Norwood, MA, USA: Artech House, 2004.

[81] M. N. Mata, J. M. Martins, and P. L. Inácio, "Impact of absorptive capacity on project success through mediating role of strategic agility: Project complexity as a moderator," *J. Innov. Knowl.*, vol. 8, no. 1, Jan. 2023, Art. no. 100327, doi: 10.1016/j.jik.2023.100327.

[82] S. L. Dorton, L. R. Maryeski, L. Ogren, I. T. Dykens, and A. Main, "A wargame-augmented knowledge elicitation method for the agile development of novel systems," *Systems*, vol. 8, no. 3, p. 27, Aug. 2020, doi: 10.3390/systems8030027.

[83] F. Aizaz, S. U. R. Khan, J. A. Khan, and A. Akhunzada, "An empirical investigation of factors causing scope creep in agile global software development context: A conceptual model for project managers," *IEEE Access*, vol. 9, pp. 109166–109195, 2021, doi: 10.1109/ACCESS.2021.3100779.

[84] H. F. Kaiser, "An index of factorial simplicity," *Psychometrika*, vol. 39, no. 1, pp. 31–36, Mar. 1974, doi: 10.1007/bf02291575.

[85] J. F. Hair, W. C. Black, B. J. Babin, and R. E. Anderson, *Multivariate Data Analysis*, 7th ed., Harlow, Essex: Pearson, 2010.

[86] C. Fornell and D. F. Larcker, "Structural equation models with unobservable variables and measurement error: Algebra and statistics," *J. Marketing Res.*, vol. 18, no. 3, pp. 382–388, Aug. 1981.

[87] J. F. Hair, R. E. Anderson, R. L. Tatham, and W. C. Black, *Multivariate Data Analysis With Readings*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1998.

[88] P. M. Bentler, "Comparative fit indexes in structural models," *Psychol. Bull.*, vol. 107, no. 2, pp. 238–246, 1990, doi: 10.1037/0033-2909.107.2.238.

[89] K. A. Bollen, *Structural Equations With Latent Variables*. New York, NY, USA: Wiley, 1989.

[90] M. W. Browne and R. Cudeck, "Alternative ways of assessing model fit," *Sociol. Methods Res.*, vol. 21, no. 2, pp. 230–258, Nov. 1992, doi: 10.1177/0049124192021002005.

[91] C. Forza and R. Filippini, "TQM impact on quality conformance and customer satisfaction: A causal model," *Int. J. Prod. Econ.*, vol. 55, no. 1, pp. 1–20, Jun. 1998, doi: 10.1016/s0925-5273(98)00007-3.

[92] D. Gefen, D. Straub, and M.-C. Boudreau, "Structural equation modeling and regression: Guidelines for research practice," *Commun. Assoc. Inf. Syst.*, vol. 4, no. 1, p. 7, 2000.

[93] K. G. Joreskog and D. Sorbom, *Advances in Factor Analysis and Structural Equation Models*. Lanham, MD, USA: Rowman & Littlefield, 1984.

[94] H. W. Marsh and D. Hocevar, "Application of confirmatory factor analysis to the study of self-concept: First- and higher order factor models and their invariance across groups," *Psychological Bulletin*, vol. 97, no. 3, p. 562, 1985.

[95] D. W. Straub, "Validating instruments in MIS research," *MIS Quart.*, vol. 13, no. 2, p. 147, Jun. 1989.

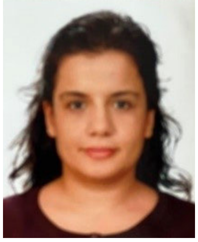
[96] R. P. Bagozzi, Y. Yi, and L. W. Phillips, "Assessing construct validity in organizational research," *Administ. Sci. Quart.*, vol. 36, no. 3, p. 421, Sep. 1991.

[97] J. F. Hair, G. T. Hult, C. Ringle, and M. Sarstedt, *A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM)*. Thousand Oaks, CA, USA: Sage, 2016.

[98] J. Henseler, C. M. Ringle, and R. R. Sinkovics, "The use of partial least squares path modeling in international marketing," *Adv. Int. Mark.*, vol. 20, pp. 277–319, Jan. 2009, doi: 10.1108/S1474-7979(2009)0000020014.

BURCU BINBOGA was born in Antalya, Türkiye, in 1990. She received the B.S. degree in industrial engineering from the University of Gazi, in 2012, and the M.S. degree in industrial engineering from Istanbul Technical University, in 2015, where she is currently pursuing the Ph.D. degree.

She held the positions of an Assistant Specialist, a Specialist, and a Senior Specialist with the IT Software Development Department, Central Securities Depository of Turkish, Istanbul Stock Exchange Group, from 2014 to 2023. She is also the Chief Expert with the Central Securities Depository of Turkish. Her research interests include agile software development, requirements engineering, business analysis, and the software testing life cycle.



CIGDEM ALTIN GUMUSSOY received the B.S. degree from Marmara University, in 2001, and the M.S. and Ph.D. degrees from Istanbul Technical University (ITU), in 2003 and 2009, respectively. She is currently an Associate Professor with the Department of Industrial Engineering, ITU. Her current research interests include project management, human–computer interaction, and technology acceptance.