

PROJECT INITIATION CAPABILITIES AND SOFTWARE PROJECT SUCCESS: A QUANTITATIVE STUDY OF SOFTWARE HOUSES IN LAHORE

Aqsa Saeed^{*1}, Dr. Muhammad Zia Aslam², Iqra Saeed³, Mahad Majeed⁴

^{*1,3}Department of Project Management, The Superior University, 17 KM Main Raiwind Road, Lahore, Punjab, Pakistan.

²Faculty of Business and Management Sciences, The Superior University, 17 KM Main Raiwind Road, Lahore, Punjab, Pakistan.

⁴National University of Sciences and Technology (NUST), Sector H-12, Islamabad, Pakistan.

^{*1}aqsasaeed6600@gmail.com, ²muhammad.zia@superior.edu.pk, ³iqra.saeed@superior.edu.pk, ⁴mahadmajeed@gmail.com.

DOI: <https://doi.org/10.5281/zenodo.22843558>

Keywords

project initiation capability; software project success; proposal quality; client communication; scope clarity; Information Processing Theory; PLS-SEM
JEL Classification: D83; L86; M15; O22; O32

Article History

Received: 01 June 2026

Accepted: 23 August 2026

Published: 08 September 2026

Copyright @Author

Corresponding Author: *

Aqsa Saeed

Abstract

Grounded in Information Processing Theory, this study examines how proposal quality and client communication during project initiation influence software project success, with scope clarity as a mediator and project complexity as a moderator. A quantitative cross-sectional survey produced 250 usable responses from professionals working in Lahore's software industry. Partial Least Squares Structural Equation Modeling (PLS-SEM) in SmartPLS 4 was used to assess the measurement and structural models. The final model explained 36.1% of the variance in scope clarity and 53.1% of the variance in software project success. Client communication was the strongest direct predictor of project success ($\beta = 0.528$, $p < .001$). Proposal quality had no significant direct effect ($\beta = -0.008$, $p = .872$) but influenced success indirectly through scope clarity ($\beta = 0.081$, $p = .003$). Client communication also had a significant indirect effect through scope clarity ($\beta = 0.124$, $p < .001$), indicating complementary partial mediation. Project complexity did not significantly moderate the focal relationships. The findings show that initiation capabilities operate through distinct information-processing pathways and support structured client communication, scope governance, and proposal-to-scope alignment as practical priorities for software firms.

1. INTRODUCTION

Software projects continue to face persistent performance problems despite advances in project management methods, agile delivery practices, development tooling, and quality assurance. Schedule slippage, cost pressure, unmet requirements, weak user acceptance, and

client dissatisfaction remain visible across information technology and software-development settings (Bojanova, 2015; Lech, 2013; Loureiro et al., 2024). These outcomes are consequential for software houses because project success affects profitability, client retention, organizational reputation, repeat business, and the ability to scale service delivery. In competitive software-service markets, unsuccessful projects

can also consume scarce technical capacity and create opportunity costs that extend beyond the focal project.

A recurring feature of software-project research is that attention often shifts quickly to execution: planning iterations, monitoring progress, managing teams, controlling quality, and responding to change. These activities are important, but execution begins from an informational and relational foundation established earlier. Before a project enters sustained delivery, the software house and client must negotiate expectations, communicate objectives, identify assumptions, define a workable scope, estimate resources, establish communication arrangements, and decide how uncertainty will be handled. Weaknesses at this stage can remain hidden until they emerge later as rework, disputed deliverables, uncontrolled scope change, missed milestones, or client dissatisfaction.

The initiation stage is therefore not simply an administrative prelude. It is a period in which incomplete and sometimes conflicting information is converted into commitments that guide later work. Jørgensen (2024) emphasizes the importance of the front-end phase in agile software development, while George et al. (2008) and Kloppenborg et al. (2006) show more generally that front-end planning and sponsor involvement shape subsequent project conditions. Requirements research similarly links the quality of early requirements and shared understanding to later project outcomes (Kamata & Tamai, 2007; Tamai & Itakura Kamata, 2009). These streams suggest that the quality of initiation deserves treatment as a distinct managerial capability rather than as a collection of isolated preliminary tasks.

This study focuses on two initiation capabilities that are particularly salient in software houses: Proposal Quality and Client Communication. Proposal Quality concerns the clarity, completeness, realism, and professional articulation of the proposed solution and associated commitments. Client Communication concerns the effectiveness of information exchange between the client and delivery

organization during initiation. The study further proposes that these capabilities matter because they create Scope Clarity, defined as a shared understanding of boundaries, deliverables, assumptions, dependencies, acceptance conditions, and change arrangements. Scope Clarity is therefore positioned as a mechanism through which initiation practices can be translated into Software Project Success.

The study is grounded in Information Processing Theory (IPT) (Galbraith, 1973, 1977). IPT argues that organizations must develop information-processing capacity that is appropriate to the uncertainty and interdependence of their tasks. Software projects are information intensive: requirements are distributed across stakeholders, technical consequences are not always obvious to clients, and new knowledge frequently emerges during discovery and delivery. Proposal development and client communication can be interpreted as information-processing mechanisms that organize, surface, and exchange information. Scope Clarity can then be understood as an information state in which key project expectations become sufficiently shared to support coordinated execution.

The Lahore software industry offers an appropriate context for examining these relationships. Lahore is one of Pakistan's major centers of software development and technology-enabled services, with firms serving domestic and international clients across web, mobile, data, and custom-software projects. The market contains organizations with different levels of project-management maturity and diverse client-engagement arrangements. Although Irfan et al. (2019) examined broader project-management capabilities in Pakistan, initiation-specific mechanisms in software houses remain comparatively underexplored. This contextual gap is important because software-service firms often operate at the boundary between sales and delivery, where proposal commitments, client communication, and scope decisions become tightly interconnected.

The study makes three contributions. First, it separates documentation-oriented and interaction-oriented initiation capabilities and

tests their direct and indirect pathways to project success. Second, it uses Scope Clarity as a theoretically meaningful mediator, allowing the analysis to explain how initiation capabilities influence outcomes rather than merely whether they correlate with them. Third, it tests whether Project Complexity, operationalized through team size, changes the strength of the focal relationships. The empirical model is tested with 250 responses using PLS-SEM, which permits simultaneous assessment of the reflective measurement model, mediation, moderation, and the relative explanatory contribution of the structural relationships.

Accordingly, the study asks the following research questions: (RQ1) What role do project initiation capabilities play in shaping Software Project Success within Lahore's software houses? (RQ1a) To what extent does Proposal Quality affect Software Project Success? (RQ1b) How does Client Communication during initiation influence Software Project Success? (RQ1c) How does Scope Clarity relate to Software Project Success? (RQ2) Does Scope Clarity mediate the relationships between Proposal Quality/Client Communication and Software Project Success? (RQ3) Does Project Complexity moderate the relationships between initiation practices and Software Project Success?

2. Literature Review and Hypothesis Development

2.1 Software Project Success

Project success has evolved from a narrow focus on the traditional iron triangle of time, cost, and scope toward a multidimensional assessment of both project-management performance and stakeholder outcomes. Shenhar et al. (1997) argued that project success includes several dimensions that may become visible at different points in time, while later work continued to distinguish efficient delivery from broader success (Lech, 2013; Serrador & Turner, 2015). For software projects, success may therefore include meeting schedule and budget expectations, satisfying users and clients, achieving intended business objectives, maintaining acceptable

quality, and supporting future relationships or continued use.

This broader view is especially relevant to software houses because a project can meet a delivery date yet still fail to create client value, or it can exceed an initial estimate while ultimately producing a highly valuable outcome. Procaccino and Verner (2006) show that software project managers evaluate success using more than a single operational criterion, and Kutsch and Maylor (2009) similarly highlight the complexity of how managers judge IT project outcomes. A multidimensional success construct is therefore appropriate for a study of initiation capability because early decisions can affect both execution efficiency and downstream client or business outcomes.

Software Project Success is treated in this study as the focal endogenous outcome. The construct captures budget performance, schedule performance, delivery quality, client-related outcomes, and business-objective attainment. This allows the structural model to evaluate whether initiation practices have explanatory relevance beyond one narrow delivery metric. It also aligns with the argument that successful software delivery involves creating a workable relationship between technical performance and stakeholder expectations rather than optimizing a single project constraint.

2.2 Proposal Quality

Proposal Quality refers to the degree to which a project proposal clearly and realistically communicates what will be delivered, how the proposed work is framed, what assumptions underpin the commitment, and what commercial or technical boundaries apply. In software-service environments, the proposal often sits at the interface between business development and delivery. It may contain features, milestones, exclusions, estimates, architecture assumptions, deliverables, payment arrangements, and representations of the intended solution. A high-quality proposal should therefore reduce ambiguity rather than merely improve sales presentation.

Prior research supports the importance of early definition and realistic commitments. Narayanan et al. (2011), in the context of outsourced software projects, connect management practices to project performance and customer satisfaction. Nasir and Sahibuddin (2011) identify requirements and estimation-related factors among the critical determinants of software initiative outcomes. George et al. (2008) further show the importance of front-end planning activities in establishing a basis for project execution. Together, these studies suggest that a proposal can function as an early information artifact that aligns commercial commitments with later delivery requirements.

The direct effect of Proposal Quality on success, however, is not theoretically inevitable. A well-written proposal can still fail to influence delivery if it is not understood by project stakeholders, if assumptions are not carried into execution, or if its contents are not translated into a clear scope. This distinction motivates the inclusion of Scope Clarity as a mediator. Proposal Quality may be valuable primarily because it improves the quality of shared project understanding. Nevertheless, because proposal content can also shape estimates, expectations, contractual commitments, and resource decisions, a direct path to project success remains theoretically plausible and is tested empirically.

2.3 Client Communication During Initiation

Client Communication during initiation refers to the clarity, timeliness, structure, and effectiveness of information exchange before sustained project execution. Software projects frequently begin with asymmetric knowledge: clients possess business knowledge and desired outcomes, while software teams possess technical knowledge about feasibility, architecture, effort, constraints, and trade-offs. Communication is the mechanism through which these distinct knowledge domains are connected. Its quality therefore influences how quickly ambiguities are surfaced, how contradictions are resolved, and how stakeholder expectations become actionable.

Communication has repeatedly been identified as a critical success factor in project and

information-systems research. Aubert et al. (2013) emphasize communication quality in enterprise-system project success, while Andersen et al. (2006) and Gemünden and Lechler (1997) position communication and stakeholder-related factors among central determinants of project outcomes. Salman et al. (2021) report a positive relationship between communication and project success in an agile software-development context. Literature reviews of ICT and IT-project success similarly identify communication and stakeholder coordination as recurring success factors (Ayat et al., 2021; Iriarte & Bayona, 2020).

The initiation stage magnifies the importance of communication because many commitments are not yet stabilized. Questions about objectives, user priorities, acceptance conditions, dependencies, access to client stakeholders, and change expectations must be resolved before they become embedded in plans or contractual positions. Effective communication can therefore produce two forms of value. First, it can improve Scope Clarity by creating shared understanding. Second, it can directly support project success through trust, responsiveness, expectation management, decision speed, and conflict resolution. This dual role is central to the proposed model.

2.4 Scope Clarity

Scope Clarity refers to the extent to which project boundaries and expectations are clearly defined, documented, and understood by relevant stakeholders. In software projects, scope is not limited to a list of features. It can include deliverables, interfaces, acceptance criteria, nonfunctional expectations, assumptions, exclusions, dependencies, validation arrangements, responsibilities, and rules for handling changes. Requirements research indicates that poor requirement quality and ambiguity are strongly connected to project difficulties (Kamata & Tamai, 2007; Tamai & Itakura Kamata, 2009). Jørgensen (2024) similarly emphasizes the importance of front-end characteristics in agile software projects.

Scope Clarity is conceptually distinct from rigidity. Agile or adaptive projects may

legitimately change their scope as learning occurs. What matters is whether stakeholders understand the current baseline, the rationale for change, and the consequences of decisions. A clear scope gives the project team a reference point for estimation, prioritization, acceptance, and change control. It also reduces the risk that client expectations and delivery-team assumptions diverge unnoticed. Accordingly, Scope Clarity should improve the ability of teams to coordinate work and evaluate progress, thereby supporting Software Project Success.

Within the proposed framework, Scope Clarity is also the mechanism that connects initiation practices with outcomes. Proposal Quality and Client Communication represent two different information-processing activities: one structures information in a relatively formal artifact, while the other exchanges and refines information through interaction. If these activities are effective, they should create a more coherent shared representation of the project. The mediator therefore converts the initiation stage from a list of practices into a theoretically specified process: information is organized and exchanged, clarity increases, and the probability of successful execution improves.

2.5 Project Complexity

Project complexity describes the extent to which a project contains interdependence, uncertainty, coordination demands, technical difficulty, and organizational or environmental variety. Bosch-Rekveltdt et al. (2011) distinguish technical, organizational, and environmental sources of complexity, while Luo et al. (2020) examine the relationship between complexity and project success. Complexity matters to information processing because more complex projects generate greater informational demands: more stakeholders must coordinate, more dependencies must be understood, and more potential interpretations of project requirements may exist.

The original survey captured both project type and team size. Project type was retained for descriptive purposes but was not combined numerically with team size because project

categories are nominal and do not possess a defensible common ordering. Project Complexity was therefore operationalized with team size as a single proxy indicator. Larger teams generally create more communication links and coordination requirements, making team size a parsimonious indicator of organizational coordination complexity. This operationalization is intentionally narrow and is treated as a limitation when interpreting the moderation result.

From an IPT perspective, the benefits of Proposal Quality and Client Communication may become stronger as information-processing demands increase. When more people are involved, structured proposals may provide a common reference and disciplined communication may become more important for preventing fragmented understanding. The model therefore tests whether team-size-based complexity strengthens the effect of initiation capabilities on Software Project Success.

2.6 Information Processing Theory

Information Processing Theory provides the central theoretical logic connecting the study variables. Galbraith (1973, 1977) argues that uncertainty creates a need for information processing and that organizations must develop mechanisms capable of meeting that need. Where information requirements exceed processing capacity, coordination becomes difficult and decision quality may deteriorate. Organizations can respond by reducing the need for information processing, increasing their capacity to process information, or creating structures that facilitate communication and coordination.

Software project initiation is a particularly suitable setting for IPT because uncertainty is concentrated at the front end. Clients may not fully understand technical constraints; development teams may not yet understand the client's operating context; business-development personnel may have knowledge that has not transferred to delivery teams; and project boundaries may still be subject to negotiation. Proposal Quality increases processing capacity by

structuring information into an explicit artifact. Client Communication increases capacity by enabling exchange, clarification, feedback, and mutual adjustment. Scope Clarity reflects the extent to which these mechanisms have produced an actionable common understanding.

The theory also provides a basis for the mediation model. Proposal Quality or Client Communication should not be expected to improve project performance simply because the activities exist. Their value depends on whether they reduce relevant uncertainty. Scope Clarity captures that reduction more directly than the initiation practices themselves. A significant indirect effect would therefore support the theoretical proposition that information-

processing mechanisms influence outcomes through the shared information state they create. The moderation test further evaluates the IPT argument that information-processing mechanisms may become more important under greater coordination complexity.

2.7 Conceptual Framework and Hypotheses

Figure 1 presents the proposed structural model. Proposal Quality and Client Communication are modeled as antecedents of Scope Clarity and as direct predictors of Software Project Success. Scope Clarity is modeled as a mediator, while Project Complexity moderates the two direct initiation-practice relationships with Software Project Success.

Proposed Conceptual Model

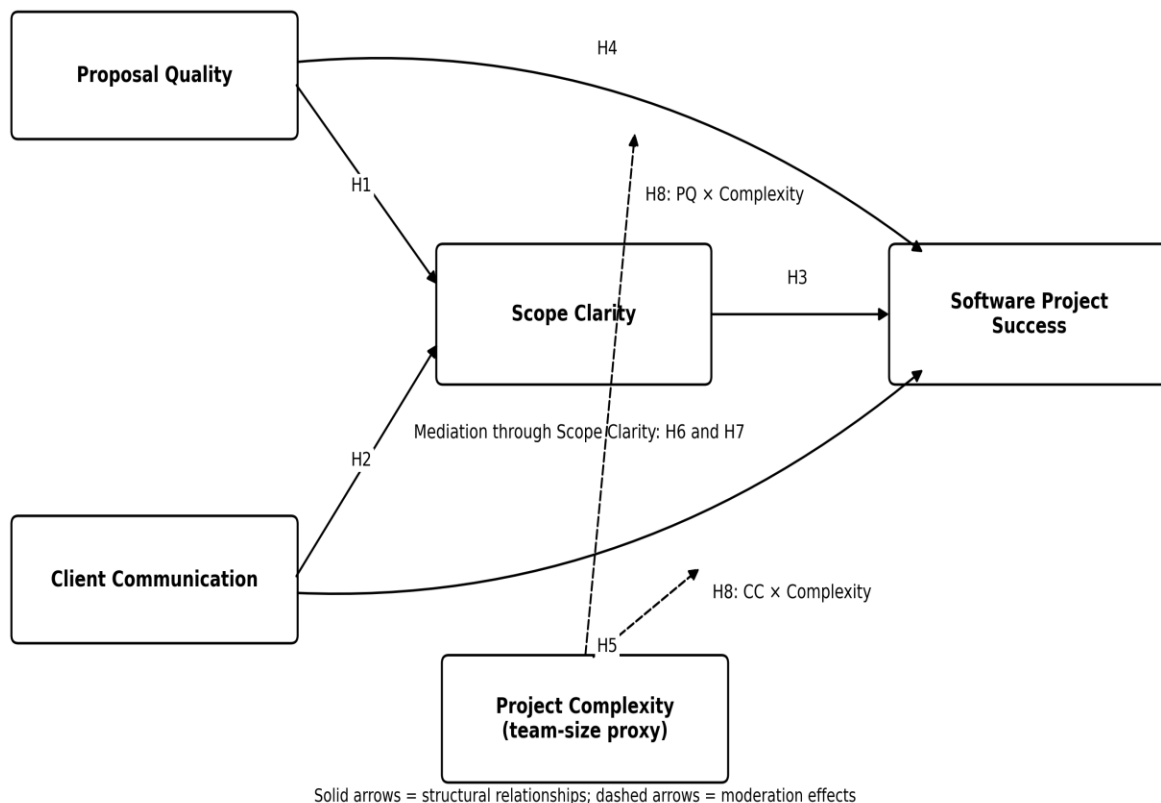


Figure 1. Proposed conceptual model connecting initiation capabilities, scope clarity, project complexity, and software project success.

H1: Proposal Quality positively relates to Scope Clarity.

H2: Client Communication during initiation positively relates to Scope Clarity.

H3: Scope Clarity positively relates to Software Project Success.

H4: Proposal Quality has a positive direct effect on Software Project Success.

H5: Client Communication during initiation has a positive direct effect on Software Project Success.

H6: Scope Clarity mediates the relationship between Proposal Quality and Software Project Success.

H7: Scope Clarity mediates the relationship between Client Communication during initiation and Software Project Success.

H8: Project Complexity moderates the relationships between initiation practices and Software Project Success, such that the relationships are stronger for more complex projects.

3. Methodology

3.1 Research Philosophy and Design

The study adopted a deductive, quantitative approach. The conceptual relationships were derived from IPT and prior project-management and software-project literature, and the hypotheses were tested using survey data. A cross-sectional design was appropriate because the study sought to estimate associations among initiation capabilities, Scope Clarity, Project Complexity, and perceived project outcomes across a broad set of software projects. The unit of analysis was the project as assessed by a professional who had direct knowledge of the project and its initiation activities.

The design emphasizes explanation rather than causal proof. The hypothesized direction follows

the temporal and theoretical logic that initiation practices precede the development of shared scope understanding and subsequent project outcomes. However, because data were collected at one point in time, the estimated relationships should be interpreted as theoretically directed associations. This limitation is considered explicitly in Section 5.4.

3.2 Population, Sampling, and Respondent Profile

The target population consisted of professionals working in software houses in Lahore, Pakistan. The software-house population includes registered and unregistered firms and does not have a single complete public sampling frame. Consequently, probability sampling was not feasible. Respondents were recruited through convenience and snowball sampling, with participation sought from individuals who had direct experience of software-project initiation and delivery. A total of 250 usable responses were obtained.

The sample included team leads and senior developers, project managers and product owners, quality-assurance professionals, business-development personnel, designers, and other employees involved in software projects. This occupational diversity is useful because initiation is a cross-functional activity involving both commercial and delivery perspectives. Team size was recorded because it was used as the proxy for Project Complexity. As shown in Table 1, almost half of the respondents (48.4%) reported project teams of four to seven members, while 32.8% worked in teams of one to three members. The remaining respondents were associated with larger teams.

Table 1. Respondent distribution by team size.

Team size	Frequency	Percentage
1-3 members	82	32.8
4-7 members	121	48.4
8-15 members	26	10.4
16+ members	21	8.4

3.3 Measurement Instrument

All multi-item constructs were measured with a five-point Likert scale ranging from 1 (strongly disagree) to 5 (strongly agree). Measurement items were adapted from established project-management, software-project, and information-systems research and reworded where necessary to fit the software-house initiation context. Proposal Quality was initially measured with six items reflecting the clarity and usefulness of the proposal as an initiation artifact. Client Communication was measured with six items reflecting structured communication, feedback, interaction, and information exchange. Scope Clarity was measured with five items capturing the clarity of boundaries, acceptance conditions, change arrangements, and project dependencies. Software Project Success was initially measured

with six items representing multiple dimensions of project outcome. Project Complexity was operationalized using the team-size proxy.

The use of adapted scales was intended to preserve theoretical correspondence while ensuring contextual relevance. The reflective specification was retained for the four multi-item constructs because the empirical analysis treated the indicators as manifestations of the underlying construct and evaluated reliability, convergent validity, and discriminant validity accordingly. During measurement-model refinement, two indicators, PQ1 and PS5, were removed because of weak outer loadings and their adverse effect on convergent validity. The remaining indicators retained adequate domain coverage while allowing the constructs to satisfy the final reliability and AVE criteria.

Table 2. Measurement constructs and sources.

Construct	Initial / retained indicators	Primary source(s)
Proposal Quality	6 (5 retained)	Narayanan et al. (2011); Nasir & Sahibuddin (2011)
Client Communication	6	Aubert et al. (2013); Gemünden & Lechler (1997); Salman et al. (2021)
Scope Clarity	5	George et al. (2008); Kamata & Tamai (2007)
Software Project Success	6 (5 retained)	Shenhar et al. (1997); Lech (2013); Serrador & Turner (2015)
Project Complexity	1 proxy indicator	Team size

3.4 Data Collection and Ethical Considerations

Data were collected using both physical and online procedures. Participation was voluntary and respondents were informed of the academic purpose of the study. CARBS granted ethical approval for the research (Approval No. 1235) on January 5, 2026. Informed consent was obtained from all participants before participation. No individually identifiable participant information is reported in this paper.

The questionnaire captured respondents' perceptions of project initiation practices and outcomes. Because the study relies on self-reported project assessments, responses represent informed perceptions rather than independently

audited project records. This approach is common in project-management research but introduces potential recall and common-source concerns, which are acknowledged in the limitations.

3.5 Data Analysis Procedure and PLS-SEM Justification

The hypothesized model was analyzed using Partial Least Squares Structural Equation Modeling (PLS-SEM) in SmartPLS 4. PLS-SEM was selected because the study seeks to explain variance in Software Project Success, compare the relative importance of several initiation-related drivers, and test mediation and moderation within an integrated latent-variable model. The

approach also enables separate assessment of the measurement model before interpretation of the structural relationships (Hair et al., 2017). The choice was therefore driven by the explanatory and prediction-oriented objectives of the research rather than by sample size alone.

The analysis proceeded in two stages. First, the reflective measurement model was assessed through outer loadings, Cronbach's alpha, rho_A, composite reliability (rho_C), average variance extracted (AVE), and the heterotrait-monotrait ratio (HTMT). Indicators with lower loadings were not removed mechanically. Deletion was considered only where it improved convergent validity and where remaining indicators continued to represent the construct's conceptual domain. Internal consistency values of approximately 0.70 or higher and AVE values of at least 0.50 were treated as acceptable. HTMT values below the conservative 0.85 criterion were used to establish discriminant validity.

Second, the structural model was assessed through inner variance inflation factor (VIF) values, coefficients of determination (R-squared), path coefficients, specific indirect effects, and interaction effects. VIF values below 3 were interpreted as indicating no material predictor-collinearity concern. Bootstrapping with 5,000 resamples was used to estimate the significance of direct, indirect, and interaction effects at $\alpha = .05$. Mediation was classified by jointly considering the significance and direction of the direct and indirect paths. Project Complexity was included as a single-item team-size proxy and SmartPLS interaction terms were used to test the moderation relationships.

For compact reporting, PQ denotes Proposal Quality, CC denotes Client Communication, PSC denotes Project Scope Clarity, and SPS denotes Software Project Success. These abbreviations are introduced here after the full construct names and are used primarily in statistical tables and path descriptions.

4. Data Analysis and Results

4.1 Measurement Model Assessment

The measurement model was evaluated before interpreting structural relationships. Initial results indicated comparatively weak loading behavior for PQ1 and PS5 and AVE values below the desired 0.50 level for Proposal Quality and Software Project Success. The two indicators were removed sequentially after considering their statistical contribution and construct coverage. The revised model produced AVE values above 0.50 for all four multi-item constructs, while internal-consistency measures remained above accepted thresholds. The final outer loadings are reported in Table 3.

Several retained indicators remained below the preferred loading level of approximately 0.708, including CC3, CC4, PQ3, PS1, PS2, and PSC5. They were retained because their loadings were generally above 0.50-0.60, the corresponding constructs achieved adequate reliability and AVE, and the indicators represented substantively relevant aspects of the constructs. In particular, the lower-loading Software Project Success items captured schedule and budget dimensions that are theoretically important and should not be eliminated solely to maximize a statistical criterion.

Table 3. Final indicator outer loadings.

Construct	Indicator	Outer loading
Client Communication	CC1	0.783
Client Communication	CC2	0.724
Client Communication	CC3	0.612
Client Communication	CC4	0.684
Client Communication	CC5	0.792
Client Communication	CC6	0.805
Proposal Quality	PQ2	0.719
Proposal Quality	PQ3	0.627
Proposal Quality	PQ4	0.810

Construct	Indicator	Outer loading
Proposal Quality	PQ5	0.736
Proposal Quality	PQ6	0.787
Software Project Success	PS1	0.590
Software Project Success	PS2	0.548
Software Project Success	PS3	0.780
Software Project Success	PS4	0.855
Software Project Success	PS6	0.778
Scope Clarity	PSC1	0.727
Scope Clarity	PSC2	0.841
Scope Clarity	PSC3	0.732
Scope Clarity	PSC4	0.725
Scope Clarity	PSC5	0.671

Table 4 reports construct reliability and convergent validity. Cronbach's alpha values ranged from 0.760 to 0.829, rho_A values ranged from 0.788 to 0.835, and composite reliability values ranged from 0.840 to 0.876. All values exceeded the 0.70 guideline. AVE values ranged

from 0.519 to 0.550, demonstrating that each multi-item construct explained more than half of the variance in its retained indicators. The measurement model therefore achieved acceptable internal consistency and convergent validity.

Table 4. Construct reliability and convergent validity.

Construct	Cronbach's alpha	rho_A	Composite reliability (rho_C)	AVE
Client Communication	0.829	0.835	0.876	0.543
Software Project Success	0.760	0.788	0.840	0.519
Proposal Quality	0.794	0.809	0.856	0.545
Scope Clarity	0.796	0.818	0.859	0.550

Discriminant validity was examined with HTMT. As shown in Table 5, every HTMT value was below 0.85. The largest value was 0.839 between Client Communication and Software Project Success. Although close to the conservative criterion, it remained within the accepted range

and is theoretically understandable because communication quality is closely connected to client-facing and coordination-related project outcomes. The remaining HTMT values were substantially lower, supporting empirical distinctiveness among the constructs.

Table 5. HTMT discriminant-validity results.

Construct pair	HTMT
Project Complexity - Client Communication	0.051
Software Project Success - Client Communication	0.839
Software Project Success - Project Complexity	0.114
Proposal Quality - Client Communication	0.503
Proposal Quality - Project Complexity	0.066
Proposal Quality - Software Project Success	0.451
Scope Clarity - Client Communication	0.652

Construct pair	HTMT
Scope Clarity - Project Complexity	0.088
Scope Clarity - Software Project Success	0.737
Scope Clarity - Proposal Quality	0.547

4.2 Structural Model Collinearity and Explanatory Power

Before testing hypotheses, inner-model VIF values were examined to determine whether overlap among structural predictors could distort the path estimates. All VIF values were low, ranging from 1.006 to 1.580 (Table 6). This indicates that the direct predictors and

interaction terms did not create a material multicollinearity problem. The results are particularly important for the Software Project Success equation because it simultaneously includes Proposal Quality, Client Communication, Scope Clarity, Project Complexity, and the two moderation terms.

Table 6. Inner-model collinearity statistics.

Predictor relationship	VIF
Client Communication → Software Project Success	1.515
Client Communication → Scope Clarity	1.220
Project Complexity → Software Project Success	1.006
Project Complexity x Client Communication → Software Project Success	1.217
Project Complexity x Proposal Quality → Software Project Success	1.231
Proposal Quality → Software Project Success	1.366
Proposal Quality → Scope Clarity	1.220
Scope Clarity → Software Project Success	1.580

The R-squared values show how much variance the model explains in the endogenous constructs. Proposal Quality and Client Communication jointly explained 36.1% of the variance in Scope Clarity. The full set of predictors explained 53.1% of the variance in Software Project

Success. These values indicate meaningful explanatory relevance for an organizational and behavioral model of software-project outcomes without requiring rigid labels such as weak, moderate, or substantial, which can vary by discipline and model context.

Table 7. Coefficients of determination.

Endogenous construct	R-squared	Variance explained
Scope Clarity	0.361	36.1%
Software Project Success	0.531	53.1%

4.3 Direct Effects and Hypothesis Testing

The bootstrapped direct effects are reported in Table 8. H1 proposed that Proposal Quality would positively relate to Scope Clarity. The relationship was positive and significant ($\beta = 0.278$, $t = 4.269$, $p < .001$), supporting H1. This result indicates that clearer and more complete

proposals are associated with stronger shared understanding of project boundaries and expectations.

H2 proposed that Client Communication would positively relate to Scope Clarity. The effect was positive, significant, and larger than the Proposal Quality effect ($\beta = 0.427$, $t = 6.849$, $p < .001$).

H2 was therefore supported. The result suggests that interactive information exchange during initiation plays a particularly important role in transforming uncertain or distributed project knowledge into a shared scope.

H3 proposed a positive relationship between Scope Clarity and Software Project Success. The path was positive and statistically significant ($\beta = 0.291$, $t = 4.546$, $p < .001$), supporting H3. This finding indicates that projects with clearer boundaries, acceptance conditions, change arrangements, and dependencies are more likely to achieve favorable project outcomes.

The two direct initiation-to-success paths produced different results. H4 proposed that Proposal Quality would directly improve Software Project Success. The coefficient was nearly zero and non-significant ($\beta = -0.008$, $t = 0.161$, $p = .872$), so H4 was not supported. H5 proposed a positive direct effect of Client Communication on Software Project Success. This path was strong and significant ($\beta = 0.528$, $t = 9.218$, $p < .001$), supporting H5. Client Communication was therefore the strongest direct predictor in the model.

Table 8. Direct-effect hypothesis testing.

Hypothesis	Path	beta	t-value	p-value	Decision
H1	PQ → PSC	0.278	4.269	< .001	Supported
H2	CC → PSC	0.427	6.849	< .001	Supported
H3	PSC → SPS	0.291	4.546	< .001	Supported
H4	PQ → SPS	-0.008	0.161	.872	Not supported
H5	CC → SPS	0.528	9.218	< .001	Supported

4.4 Mediation Analysis

Mediation was assessed through the bootstrapped specific indirect effects rather than through separate causal-step or Sobel procedures. The indirect effect of Proposal Quality on Software Project Success through Scope Clarity was positive and significant ($\beta = 0.081$, $t = 2.982$, $p = .003$). Because the corresponding direct effect of Proposal Quality on Software Project Success was non-significant, the pattern is classified as indirect-only mediation. H6 was supported. The result suggests that Proposal Quality contributes to project outcomes insofar as it creates greater Scope Clarity.

The specific indirect effect of Client Communication on Software Project Success through Scope Clarity was also positive and significant ($\beta = 0.124$, $t = 4.006$, $p < .001$). Because the direct Client Communication effect remained positive and significant, the pattern represents complementary partial mediation. H7 was supported. Client Communication therefore appears to influence project success through two routes: one operating through improved Scope Clarity and another capturing additional benefits of communication such as responsiveness, trust, coordination, and expectation management.

Table 9. Mediation results.

Hypothesis	Indirect path	Indirect effect	t-value	p-value	Mediation type	Decision
H6	PQ → PSC → SPS	0.081	2.982	.003	Indirect-only	Supported
H7	CC → PSC → SPS	0.124	4.006	< .001	Complementary partial	Supported

4.5 Moderation Analysis

H8 proposed that Project Complexity would strengthen the effects of Proposal Quality and

Client Communication on Software Project Success. Neither interaction effect reached statistical significance. The Proposal Quality x

Complexity effect was small and positive ($\beta = 0.043$, $t = 0.873$, $p = .383$), while the Client Communication x Complexity effect was small and negative ($\beta = -0.045$, $t = 0.753$, $p = .451$). The direct effect of Project Complexity on Software Project Success was also non-significant ($\beta = -0.052$, $t = 1.222$, $p = .222$). H8 was therefore not supported.

The unsupported moderation result indicates that the observed effects of initiation capabilities did not vary systematically across the team-size

categories represented in the study. This does not establish that complexity is irrelevant in software projects. Instead, it shows that team-size-based complexity did not change the focal relationships in this sample. The result should be interpreted in light of the deliberately narrow complexity measure, which does not capture technical novelty, stakeholder diversity, geographic dispersion, requirement volatility, or integration intensity.

Table 10. Project Complexity and moderation results.

Relationship	beta	t-value	p-value	Decision
Project Complexity -> SPS	-0.052	1.222	.222	Not significant
PQ x Complexity -> SPS	0.043	0.873	.383	Not supported
CC x Complexity -> SPS	-0.045	0.753	.451	Not supported

4.6 Summary of Hypothesis Testing

Table 11. Summary of hypothesis decisions.

Hypothesis	Statement	Decision
H1	Proposal Quality positively relates to Scope Clarity.	Supported
H2	Client Communication during initiation positively relates to Scope Clarity.	Supported
H3	Scope Clarity positively relates to Software Project Success.	Supported
H4	Proposal Quality has a positive direct effect on Software Project Success.	Not supported
H5	Client Communication during initiation has a positive direct effect on Software Project Success.	Supported
H6	Scope Clarity mediates Proposal Quality -> Software Project Success.	Supported (indirect-only)
H7	Scope Clarity mediates Client Communication -> Software Project Success.	Supported (complementary partial)
H8	Project Complexity moderates the initiation-practice -> success relationships.	Not supported

5. Discussion and Recommendations**5.1 Interpretation of the Main Findings**

The results show that project initiation capabilities influence Software Project Success through different information-processing mechanisms rather than through a single uniform pathway. Client Communication emerged as the strongest direct predictor of success, Scope Clarity had a significant positive effect, and Proposal Quality influenced success only indirectly through Scope Clarity. This pattern is theoretically informative because it differentiates the value of a formal initiation artifact from the value of ongoing interaction. Both matter, but they matter in different ways.

The Proposal Quality result is especially important. H1 was supported, indicating that better proposals contribute to clearer project scope, but H4 was not supported, showing that Proposal Quality has essentially no independent direct effect once Scope Clarity and the other predictors are taken into account. The significant indirect-only mediation in H6 resolves this apparent tension. Proposal Quality is valuable when it helps convert commercial and technical information into a shared project baseline. A proposal that is detailed but disconnected from delivery, or that is not actively translated into scope understanding, may have little independent effect on later success.

This interpretation is consistent with front-end planning and requirements research. George et al. (2008) emphasize that front-end planning is valuable because it establishes conditions for later execution, while Kamata and Tamai (2007) and Tamai and Itakura Kamata (2009) connect requirement quality with project outcomes. In outsourced software projects, Narayanan et al. (2011) similarly show that management practices can shape performance and satisfaction. The present study adds a mechanism to this literature: Proposal Quality appears to matter through the clarity it creates rather than as a standalone success factor.

Client Communication produced a broader effect. It significantly predicted Scope Clarity and also exerted a strong direct effect on Software Project Success. The complementary partial

mediation result indicates that communication has value both because it clarifies scope and because it contributes to outcomes through other relational and coordination processes. This is consistent with Aubert et al. (2013), Salman et al. (2021), and the broader project-success literature identifying communication as a recurring determinant of effective delivery (Andersen et al., 2006; Ayat et al., 2021; Iriarte & Bayona, 2020). The magnitude of the Client Communication coefficient is also managerially significant. Software houses often treat communication as an informal interpersonal skill that varies by project manager or account manager. The results suggest that it should instead be treated as an organizational capability with structures, standards, and accountability. Formal kick-off meetings, a defined client-side decision owner, a centralized communication channel, agreed response times, documented decisions, and regular demonstrations are not merely administrative preferences. They increase the organization's capacity to process information and maintain alignment over the life of the project.

Scope Clarity itself significantly improved Software Project Success, supporting the idea that shared understanding is a critical bridge between initiation and execution. The finding should not be interpreted as an argument for inflexible scope. In agile development, learning and adaptation are expected. What matters is that the current state of agreement is visible and that changes are consciously managed. Clear deliverables, acceptance criteria, assumptions, dependencies, and change arrangements provide a reference that enables adaptation without creating uncontrolled ambiguity.

The unsupported moderation effect provides a useful boundary finding. The model did not show that larger teams receive stronger benefits from Proposal Quality or Client Communication. One interpretation is that disciplined initiation practices are broadly useful across team sizes: small teams still need clear scope and responsive clients, while large teams cannot rely on scale alone to make initiation practices more consequential. A second

interpretation is methodological: team size captures coordination load but not the full multidimensional nature of project complexity. A richer measure may reveal contingencies that the present proxy cannot detect.

5.2 Theoretical Contributions

The first theoretical contribution is the application of IPT to software-project initiation. The theory has a natural fit with project environments characterized by uncertainty and interdependence, but the present model specifies concrete initiation mechanisms. Proposal Quality represents structured codification of project information, Client Communication represents interactive information exchange, and Scope Clarity represents the shared information state that results when those mechanisms function effectively. The significant mediation effects therefore provide empirical support for the theoretical sequence from information-processing capability to reduced ambiguity and then to improved project outcomes.

Second, the study distinguishes documentation-driven and interaction-driven routes to project success. Proposal Quality operates through indirect-only mediation, whereas Client Communication operates through complementary partial mediation. This difference advances the literature beyond undifferentiated lists of critical success factors. It shows that two practices associated with initiation can occupy different causal positions in the model. Formal documentation primarily appears to create value by improving clarity; interactive communication additionally creates relational and coordination benefits that remain important even after Scope Clarity is controlled.

Third, the study contributes to the project-success literature by locating a measurable mechanism at the boundary between sales and delivery. In software houses, initiation may involve people from business development, project management, technical leadership, and the client organization. Information can be lost as work moves from proposal preparation to execution. The mediation results imply that the quality of this transition matters theoretically: information-

processing practices must produce shared scope understanding if their potential value is to reach project outcomes.

Fourth, the study provides context-specific evidence from Lahore's software industry. Irfan et al. (2019) show the relevance of project-management capabilities in Pakistan at a broader level; the present study narrows the focus to initiation and identifies which elements matter most within a software-house context. This contributes evidence from a developing software-services economy where client relationships, outsourcing, and organizational maturity may differ from settings that dominate the existing project-management literature.

5.3 Practical Implications and Recommendations

The strongest practical implication is that software houses should institutionalize client communication at initiation. A minimum communication protocol should identify stakeholders and decision makers, establish a primary communication platform, define response expectations, document key decisions, and specify escalation arrangements. The goal is not to increase meeting volume but to reduce information latency and ambiguity. Organizations can monitor unresolved questions, delayed approvals, conflicting client instructions, and decision turnaround as early warning indicators of project risk.

Second, proposal preparation should be connected directly to scope governance. Because Proposal Quality did not produce a direct success effect, firms should avoid assessing proposals only on commercial persuasiveness, visual presentation, or the speed with which they enable a sale. Proposal content should be reviewed by delivery stakeholders and translated into an execution baseline. Assumptions, exclusions, milestones, estimates, technical constraints, responsibilities, and commercial commitments should be reconciled with the scope, backlog, acceptance criteria, or statement of work used by the delivery team.

Third, software houses should introduce a lightweight initiation gate before major execution

commitments are made. The gate should confirm that the project objective is understood, key stakeholders are identified, communication arrangements are active, critical dependencies and assumptions are visible, acceptance criteria are sufficiently defined, and major feasibility questions have been addressed. The gate can be scaled to project needs: a small fixed-scope project may require a concise checklist, while an uncertain or technically novel engagement may require discovery, prototyping, or phased estimation.

Fourth, Scope Clarity should be treated as a living governance condition rather than a one-time document. Teams should maintain explicit visibility of what is included, what is excluded, how changes are approved, who can make acceptance decisions, and which dependencies may affect delivery. In agile projects, this can be accomplished through clearly defined backlog items, acceptance criteria, product goals, decision logs, and release boundaries rather than through excessive upfront specification.

Fifth, organizations should strengthen sales-to-delivery handover. Business-development personnel often accumulate valuable contextual knowledge during client conversations, while technical teams may receive only the final proposal. A formal handover should therefore communicate client objectives, commercial commitments, unresolved assumptions, technical risks, dependencies, stakeholder sensitivities, and decision history. Cross-functional review can reduce the probability that proposal promises and delivery realities diverge after project start.

Finally, management should not reserve disciplined initiation practices only for large teams. The moderation results provide no evidence that team size changes the value of the focal capabilities. A minimum standard for communication, scope clarity, and proposal alignment should therefore apply across projects, with the depth of documentation and governance adjusted proportionately rather than eliminated for smaller projects.

5.4 Limitations

Several limitations qualify the conclusions. First, the cross-sectional design does not establish causality. The theoretical ordering is plausible, but longitudinal data collected at initiation and project completion would provide stronger evidence about temporal effects. Second, the study relies on self-reported perceptions. Respondents assessed both initiation practices and project outcomes, which may introduce recall error, social-desirability bias, or common-source effects. Objective schedule, budget, defect, deployment, or client-retention records were not used to validate perceived project success.

Third, one respondent represented each project. Different stakeholders can perceive the same initiation process differently, and disagreement between the client, project manager, technical lead, and business-development personnel may itself be an important predictor of project risk. Fourth, convenience and snowball sampling were used because a complete sampling frame was unavailable. The findings therefore should not be interpreted as statistically representative of all software houses in Lahore or Pakistan.

Fifth, Project Complexity was measured with team size as a single proxy. Team size captures coordination demands but not technical novelty, requirement volatility, stakeholder diversity, integration complexity, regulatory exposure, or geographical distribution. This limitation is particularly important because the moderation hypothesis was not supported. Sixth, two indicators were removed during measurement-model refinement. Although the final constructs met reliability and validity criteria, future scale development should improve these items rather than assuming that the underlying content is unimportant.

Finally, the model was assessed primarily for explanation within the sample. A separate out-of-sample predictive assessment such as PLSpredict or other prediction-oriented validation was not included. Future research can therefore strengthen both causal and predictive claims by combining longitudinal designs, objective outcome data, and out-of-sample evaluation.

5.5 Future Research Directions

Future studies should replicate the model in other Pakistani technology centers, including Karachi and Islamabad/Rawalpindi, and in international outsourcing contexts. Comparative studies could test whether cultural distance, contract type, client location, organizational maturity, or development methodology changes the relationships. A larger multi-city design would also allow researchers to examine firm-level differences and potentially use multilevel modeling.

Project Complexity should be represented with a validated multidimensional measure. Technical complexity, organizational complexity, stakeholder complexity, environmental uncertainty, integration density, requirement volatility, and team distribution can be assessed separately. This would provide a more rigorous test of the IPT proposition that information-processing mechanisms become more valuable as information demands increase.

Future research should also use multiple informants and objective project records. Matching client and provider assessments could reveal whether alignment itself mediates project outcomes. Longitudinal designs could measure Proposal Quality and Client Communication at initiation, Scope Clarity after discovery or early planning, and Software Project Success at later milestones or closure. Such temporal separation would strengthen causal inference and reduce reliance on one-time retrospective perception.

Additional mediators may also be examined. Trust, stakeholder alignment, requirement quality, project-team readiness, and sales-to-delivery knowledge transfer are plausible mechanisms linking initiation practices to outcomes. Potential moderators include contract type, agile maturity, client experience, prior relationship history, team distribution, and requirement volatility. These extensions would help establish when different initiation capabilities are most valuable and through which processes they operate.

6. Conclusion

This study examined how Proposal Quality and Client Communication during project initiation influence Software Project Success in Lahore-based software houses. The results show that initiation capabilities do not operate through a single pathway. Client Communication is the strongest direct predictor of Software Project Success and also improves success indirectly through Scope Clarity. Proposal Quality, by contrast, has no significant direct effect once the full model is considered; its contribution is transmitted through Scope Clarity. These findings position Scope Clarity as a central information-processing outcome that connects early project practices with later delivery performance.

The model explained 36.1% of the variance in Scope Clarity and 53.1% of the variance in Software Project Success. Project Complexity, measured through team size, did not directly predict success and did not significantly moderate the focal relationships. The practical implication is that software houses should establish structured client communication, translate proposal commitments into a shared scope baseline, and apply consistent initiation governance across projects. The theoretical implication is that information-processing mechanisms are valuable when they create shared, actionable understanding and when they also support continuing coordination between project stakeholders.

Conflict of Interest

The authors declare that they have no competing financial, personal, or professional interests that could have influenced the research reported in this paper.

Ethics Approval and Consent to Participate

CARBS approved the study (Approval No. 1235) on January 5, 2026. All participants provided informed consent before participation through physical or online procedures.

Availability of Data and Materials

The dataset cannot be shared publicly because of confidentiality obligations to participants and the underlying source data. Requests for additional methodological information may be directed to the corresponding author.

Author Contributions

Aqsa Saeed led conceptualization, methodology, data analysis, interpretation, and manuscript preparation. Dr. Muhammad Zia Aslam supervised the study and provided methodological guidance and critical review. Iqra Saeed contributed to the literature review and manuscript review. Mahad Majeed supported data collection and manuscript review. All authors reviewed and approved the final manuscript.

Declaration of Generative AI Assistance

ChatGPT (OpenAI) was used to assist with manuscript restructuring, language editing, and formatting. The authors reviewed and verified the manuscript content, statistical results, interpretations, and references and take full responsibility for the submitted work. Generative AI was not used to collect the study data or execute the SmartPLS analysis.

Acknowledgement

The authors gratefully acknowledge the study participants for contributing their time and responses. No external research funding was received for this study.

REFERENCES

- Andersen, E. S., Birchall, D., Jessen, S. A., & Money, A. H. (2006). Exploring project success. *Baltic Journal of Management*, 1(2), 127-147. <https://doi.org/10.1108/17465260610663854>
- Aubert, B., Hooper, V., & Schnepel, A. (2013). Reworking the role of the quality of communication in the ERP project success. *American Journal of Business*, 28(1), 64-85.

- Ayat, M., Imran, M., Ullah, A., & Kang, C. W. (2021). Analysis of current trends and prioritisation of success factors: A structured literature review of ICT projects. *International Journal of Managing Projects in Business*, 14(3), 652-679.
- Bamgbose, O. A., Ogunbayo, B. F., Aigbavboa, C. O., & Ogundipe, K. E. (2024). A systematic review of client satisfaction and success factors in BIM-enabled projects. *Engineering Proceedings*, 76(1), 33. <https://doi.org/10.3390/engproc2024076033>
- Bojanova, I. (2015). Failure of IT projects: What can management learn? *IT Professional*, 17(6), 8-11.
- Bosch-Rekveltdt, M., Jongkind, Y., Mooi, H., Bakker, H., & Verbraeck, A. (2011). Grasping project complexity in large engineering projects: The TOE (technical, organizational and environmental) framework. *International Journal of Project Management*, 29(6), 728-739. <https://doi.org/10.1016/j.ijproman.2010.07.008>
- Galbraith, J. R. (1973). Designing complex organizations. Addison-Wesley.
- Galbraith, J. R. (1977). Organization design. Addison-Wesley.
- Garousi, V., Tarhan, A., Pfahl, D., Coşkunçay, A., & Demirörs, O. (2019). Correlation of critical success factors with success of software projects: An empirical investigation. *Software Quality Journal*, 27(1), 429-493. <https://doi.org/10.1007/s11219-018-9419-5>
- Gemünden, H. G., & Lechler, T. (1997). Project management: Success factors: The critical few-An empirical study. Paper presented at the International Project Management Association World Congress.
- George, R., Bell, L. C., & Back, W. E. (2008). Critical activities in the front-end planning process. *Journal of Management in Engineering*, 24(2), 66-74. [https://doi.org/10.1061/\(ASCE\)0742-597X\(2008\)24:2\(66\)](https://doi.org/10.1061/(ASCE)0742-597X(2008)24:2(66))

- Hair, J. F., Hult, G. T. M., Ringle, C. M., & Sarstedt, M. (2017). A primer on partial least squares structural equation modeling (PLS-SEM) (2nd ed.). Sage.
- Irfan, M., Hassan, M., & Hassan, N. (2019). The effect of project management capabilities on project success in Pakistan: An empirical investigation. *IEEE Access*, 7, 39417-39431.
<https://doi.org/10.1109/ACCESS.2019.2906851>
- Iriarte, C., & Bayona, S. (2020). IT projects success factors: A literature review. *International Journal of Information Systems and Project Management*, 8(2), 49-78.
<https://doi.org/10.12821/ijispm080203>
- Jørgensen, M. (2024). A systematic literature review on characteristics of the front-end phase of agile software development projects and their connections to project success. *Journal of Systems and Software*, 216, 112155.
<https://doi.org/10.1016/j.jss.2024.112155>
- Kamata, M. I., & Tamai, T. (2007). What is the relationship between requirements quality and project success or failure? In 15th IEEE International Requirements Engineering Conference (RE 2007) (pp. 69-78). IEEE Computer Society.
- Kloppenborg, T. J., Tesch, D., & Heitkamp, T. (2006). A research study on the involvement of the sponsor in the initiation of the project. *Project Management Journal*, 37(3), 29-40.
- Kutsch, E., & Maylor, H. (2009). Failure to success: An exploration of what managers use to gauge the result of IT projects. *International Journal of Manufacturing Technology and Management*, 16(3), 265-282.
- Lech, P. (2013). Time, money, functionality?-IT project success criteria updated. *Information Systems Management*, 30(3), 263-275.
- Loureiro, E., Gomes, B., Varajão, J., & Silva, C. (2024). Information systems project success surveys-Insights from the last 30 years. *Heliyon*, 10(23), e40619.
<https://doi.org/10.1016/j.heliyon.2024.e40619>
- Luo, L., Zhang, L., & He, Q. (2020). Associating a project complexity with project success: Hybrid SEM-FCM approach. *Engineering, Construction and Architectural Management*, 27(9), 2591-2614.
- Müller, R., & Turner, R. (2007). The influence of project managers on project success criteria and project success by type of project. *European Management Journal*, 25(4), 298-309.
<https://doi.org/10.1016/j.emj.2007.06.003>
- Narayanan, S., Balasubramanian, S., & Swaminathan, J. M. (2011). Managing outsourced software projects: An analysis of project performance and customer satisfaction. *Production and Operations Management*, 20(4), 508-521.
<https://doi.org/10.1111/j.1937-5956.2010.01162.x>
- Nasir, M. H. N., & Sahibuddin, S. (2011). Critical success factors of software initiatives: A comparative study. *Scientific Research and Essays*, 6(10), 2174-2186.
- Procaccino, J. D., & Verner, J. M. (2006). Project success and software project managers: An exploratory study. *Journal of Systems and Software*, 79(11), 1541-1551.
- Salman, A., Jaafar, M., Malik, S., Mohammad, D., & Muhammad, S. A. (2021). An empirical investigation of the impact of communication and employee motivation on project success using Agile framework and its effect on the software development business. *Business Perspectives and Research*, 9(1), 46-61.
- Serrador, P., & Turner, R. (2015). The relationship between project success and project efficiency. *Project Management Journal*, 46(1), 30-39.
<https://doi.org/10.1002/pmj.21468>

- Shenhav, A. J., Levy, O., & Dvir, D. (1997). Mapping the dimensions of project success. *Project Management Journal*, 28(2), 5-13.
- Tamai, T., & Itakura Kamata, M. (2009). Effects of requirements quality on success or failure of projects. In K. Lyytinen et al. (Eds.), *Design requirements workshop* (Lecture Notes in Business Information Processing, Vol. 14, pp. 258-275). Springer.
- Zaveri, T., Patel, M., Shah, V., Patel, N., Shastri, S., & Kiran, M. B. (2022). Difficulties in the start of projects. Paper presented at the Australian International Conference on Industrial Engineering and Operations Management, Sydney, Australia.

