

Design Science Research Approach for Quantifying Client Readiness in Custom Software Projects (CRAM)

**Dr. MohamedElfatih Abd Elrahman Mohamed Ali¹,
Prof. Eltayeb Elsamani Abdelgabar Elsamani²**

¹Information technology, University of Holy Quran and Taseel of Science

²Al-Neelain University

Abstract

Client qualification in custom software development is often performed using informal and experience-driven practices, leading to inefficient pre-sales efforts, poor opportunity prioritization, and increased project risk. Existing qualification frameworks such as BANT, MEDDIC, and SPIN primarily emphasize commercial considerations and do not adequately address the operational, technical, and collaborative requirements associated with software-intensive projects.

This study addresses this gap by developing and validating the Client Readiness and Alignment Model (CRAM), a structured multi-dimensional framework for assessing client readiness prior to project initiation. CRAM integrates commercial, technical, operational, and behavioural dimensions into a unified weighted scoring mechanism designed to support objective and consistent pre-sales decision-making.

Following the Design Science Research (DSR) methodology, the study develops CRAM as both a conceptual model and a web-based decision-support artifact. The framework was evaluated in an industrial context using a dataset of more than 50 prospective clients across software development, system integration, and automation projects.

The results demonstrate that CRAM effectively differentiates levels of client readiness, improves consistency in qualification decisions, and reduces inefficient engagement with low-readiness prospects. The study contributes both theoretically and practically by reframing client qualification as a measurable readiness construct and operationalizing it into a deployable decision-support system for software engineering environments.

Keywords: Client Readiness, Pre-Sales Evaluation, Design Science Research, Decision Support Systems, Software Project Risk, Multi-Criteria Assessment, CRAM.

1. Introduction

Custom software development projects are inherently characterized by uncertainty, evolving requirements, and strong dependence on continuous client involvement (Boehm & Turner, 2004; Standish Group, 2020). Unlike standardized software products, custom development initiatives require sustained collaboration between service providers and clients throughout requirements elicitation, validation, implementation, and deployment phases. However, many prospective clients enter software engagements without sufficient

organizational preparedness, requirements clarity, technical understanding, or operational commitment. This often leads to inefficient pre-sales activities, unstable project scopes, communication breakdowns, and increased delivery risk.

In practice, client qualification in software consulting and automation environments is frequently conducted through informal judgment and sales experience rather than structured evaluation mechanisms. Existing qualification frameworks such as BANT, MEDDIC, and SPIN provide useful commercial screening capabilities, yet they primarily focus on budget, authority, sales opportunity progression, and purchasing intent. These approaches insufficiently capture software-specific readiness factors such as requirements maturity, technical compatibility, operational preparedness, and collaborative engagement quality.

This reveals a significant gap between traditional sales qualification approaches and the practical realities of software-intensive projects. While previous studies have extensively examined requirements engineering, agile collaboration, and software project success factors, limited attention has been given to the development of structured pre-engagement mechanisms capable of quantitatively assessing client readiness before substantial project commitment occurs.

To address this gap, this study proposes the Client Readiness and Alignment Model (CRAM), a structured multi-dimensional framework designed to evaluate prospective client readiness in custom software development and automation contexts. The model integrates commercial, technical, operational, and behavioural dimensions into a weighted scoring mechanism supported by a web-based decision-support system.

The study follows the Design Science Research (DSR) methodology to design, implement, and evaluate CRAM as both a conceptual framework and a deployable artifact intended to improve pre-sales efficiency, decision consistency, and project risk management.

2. Literature Review

2.1 Classical Qualification Frameworks and Their Relevance

Several qualification frameworks have been widely adopted in sales and business development to assess prospect suitability and prioritize engagement opportunities. Among the most recognized models are BANT, MEDDIC, and SPIN, each designed to support structured prospect evaluation and improve sales effectiveness.

The BANT framework evaluates prospects based on four primary dimensions: Budget, Authority, Need, and Timeline (Rackham, 1988). Its simplicity and rapid applicability make it suitable for early-stage commercial qualification, particularly in transactional sales environments. MEDDIC extends this approach by incorporating additional dimensions such as Metrics, Economic Buyer, Decision Criteria, Decision Process, Identify Pain, and Champion, thereby providing a more comprehensive structure for managing complex enterprise sales processes (Dibble, 2011). SPIN Selling emphasizes consultative questioning strategies based on Situation, Problem, Implication, and Need-payoff interactions to uncover client needs and improve sales communication (Rackham, 1988).

These frameworks support standardization in sales activities, reduce arbitrary decision-making, and improve consistency in opportunity assessment. However, they were not specifically designed for custom software development or automation environments. Their primary focus remains commercial viability rather than delivery feasibility.

In software-intensive projects, successful engagement depends not only on budget availability or purchasing authority, but also on requirements maturity, technical compatibility, operational preparedness, stakeholder responsiveness, and sustained collaboration capability. Consequently, a prospect may satisfy traditional sales qualification criteria while still representing a high-risk engagement due to unclear workflows, insufficient organizational readiness, poor technical alignment, or limited participation capacity.

2.2 Recent Related Studies

Recent research increasingly emphasizes the importance of client-side and organizational factors in determining software project success. In agile and collaborative development environments, project outcomes are strongly influenced not only by technical implementation capabilities but also by the quality of stakeholder engagement, communication practices, and requirements maturity.

Barros and Tam (2024) identified customer involvement as one of the most significant human-related critical success factors in agile software development projects. Similarly, Qayyum et al. (2024) highlighted the importance of communication effectiveness and stakeholder engagement in improving team performance and software delivery success.

Requirements engineering literature also reinforces the importance of early organizational preparedness. Amin and Shahzad (2024) demonstrated that inadequate requirements definition, weak communication, and limited customer engagement significantly contribute to project failure in large-scale software systems. Likewise, Chen et al. (2024) showed that structured validation mechanisms can significantly improve requirements consistency and reduce project uncertainty.

Hoy and Xu (2023), through a systematic literature review of agile requirements engineering challenges, concluded that software project difficulties frequently emerge from interactions among business conditions, project management constraints, and agile implementation practices.

Collectively, these studies demonstrate that successful software project delivery depends heavily on client readiness, stakeholder collaboration, communication quality, and organizational preparedness. However, most existing approaches focus on improving project execution after engagement has already commenced rather than evaluating whether a prospective client is sufficiently prepared before project initiation.

2.3 Critical Synthesis and Research Gap

The reviewed literature reveals several important observations. First, early-stage uncertainty remains a persistent challenge in software development projects, particularly when requirements are unclear or unstable. Second, client involvement and communication quality are consistently identified as critical determinants of project success. Third, organizational preparedness and stakeholder responsiveness significantly influence delivery feasibility and implementation outcomes.

Although previous studies have extensively addressed requirements engineering practices, stakeholder communication, and software project success factors, these contributions largely concentrate on improving project execution after formal engagement has already begun. Similarly, existing sales qualification frameworks such as BANT, MEDDIC, and SPIN primarily evaluate commercial viability without adequately addressing software-specific readiness conditions.

As a result, a significant research gap remains in the development of structured pre-engagement mechanisms capable of quantitatively assessing client readiness prior to project commitment. Existing approaches insufficiently integrate commercial, technical, operational, and behavioural dimensions into a unified decision-support framework tailored to software and automation environments.

This study addresses this gap through the development of CRAM, which conceptualizes client qualification as a measurable readiness construct rather than a purely experience-driven sales judgment process.

2.4 Positioning of CRAM

CRAM is positioned as a domain-specific pre-sales assessment framework designed for software development and automation projects. Unlike traditional qualification models that primarily emphasize commercial indicators, CRAM integrates operational readiness, technical compatibility, communication quality, stakeholder engagement, and organizational preparedness into a unified evaluation structure.

The novelty of CRAM lies not only in the integration of these dimensions, but also in their operationalization through a weighted quantitative scoring mechanism supported by a web-based decision-support system. This enables organizations to systematically evaluate prospective clients, improve prioritization decisions, and reduce reliance on subjective intuition during pre-sales activities.

Accordingly, CRAM contributes both conceptually and practically. Conceptually, it extends traditional qualification approaches toward software-specific readiness assessment. Practically, it delivers a deployable artifact capable of supporting structured and repeatable decision-making in real-world software engineering environments.

3. Methodology

3.1 Research Approach

This study adopts the Design Science Research (DSR) methodology, which is widely recognized for addressing complex real-world problems through the systematic design, development, and evaluation of innovative artifacts (Hevner et al., 2004; Peffers et al., 2007). DSR is particularly suitable for this research because the study aims not only to analyze the problem of inefficient client qualification but also to construct and validate a practical solution.

The artifact developed in this study consists of two integrated components:

1. The conceptual CRAM framework that formalizes client readiness assessment dimensions and scoring logic.
2. A web-based decision-support implementation that operationalizes the evaluation process within practical pre-sales environments.

The research follows the six-step DSR process proposed by Peffers et al. (2007), ensuring methodological rigor and traceability between problem identification, artifact development, and evaluation.

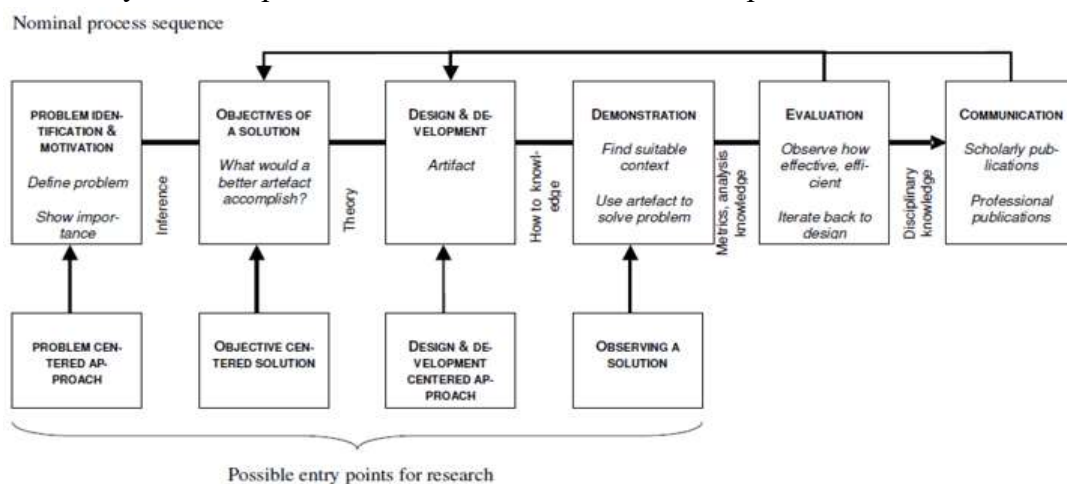


Figure 1. DSR methodology process model

3.2 Design and Development of CRAM

The CRAM framework was designed as a structured evaluation model consisting of seven readiness dimensions:

Table 1. CRAM Dimensions and Weights

No.	Dimension	Weight
1	Intent and Actual Need	20%
2	Budget and Financial Readiness	25%
3	Timeline	10%
4	Operational Readiness	15%
5	Technical Fit	10%
6	Communication and Engagement	10%
7	Decision Authority	10%

The weighting structure of CRAM was established through iterative expert-informed evaluation derived from practical experience in software consulting and pre-sales assessment activities. The weights were refined during pilot application of the model to reflect the relative influence of each dimension on project feasibility, engagement quality, and delivery risk.

3.3 Measurement Model and Scoring Mechanism

Each indicator is evaluated using a three-point Likert scale:

- A. 0 = Low / Not Available
- B. 1 = Moderate
- C. 2 = High / Strong

3.3.1 Dimension Score

$$D_i = \frac{\sum_{j=1}^n I_j}{2n}$$

Where:

- A. D_i = normalized score of dimension i
- B. I_j = indicator score
- C. n = number of indicators within the dimension

3.3.2 Overall Readiness Score

$$R = \sum_{i=1}^m w_i D_i$$

Where:

- A. R = overall readiness score
- B. w_i = weight assigned to dimension i
- C. D_i = normalized score of dimension i

3.3.3 Classification Model

Table 2. Client Readiness Classification.

Readiness Score	Classification	Recommended Engagement Strategy
0–40%	Non-Serious	Minimal engagement
41–70%	Conditional	Controlled engagement
71–100%	Ready	High-priority engagement

4. Results

The CRAM model was evaluated using a dataset of 50 prospective clients across custom software development, system integration, and business automation projects. The evaluation results revealed a clear distribution of readiness levels, with 30% of clients classified as ready, 42% as conditional, and 28% as non-serious.

The findings demonstrate that CRAM possesses strong discriminatory capability in differentiating clients based on readiness characteristics. High-readiness clients consistently exhibited clearer requirements, stronger organizational preparedness, higher engagement responsiveness, and more structured decision-making processes. In contrast, low-readiness clients were characterized by ambiguous requirements, limited commitment, inconsistent communication, and unclear operational expectations.

The implementation of CRAM contributed to improved pre-sales efficiency by reducing unnecessary engagement with low-readiness prospects and enabling more effective prioritization of opportunities. Additionally, the model enhanced consistency and standardization in qualification decisions, thereby reducing reliance on subjective judgment and experience-driven evaluation practices.

5. Research Contributions

This study contributes to both research and practice in several ways.

From a theoretical perspective, the study extends traditional client qualification approaches by introducing client readiness as a measurable multi-dimensional construct tailored to software development and automation environments.

From a methodological perspective, the study demonstrates the applicability of Design Science Research (DSR) for developing and evaluating practical decision-support artifacts in software engineering contexts. From a practical perspective, the study delivers a deployable web-based decision-support system capable of supporting structured pre-sales assessment, improving opportunity prioritization, and reducing inefficient engagement with low-readiness prospects.

6. Conclusion

This study introduced the Client Readiness and Alignment Model (CRAM) as a structured framework for improving client qualification in custom software development and automation environments. The proposed model transforms traditionally informal and experience-driven qualification practices into a measurable, multi-dimensional readiness assessment process integrating commercial, technical, operational, and behavioural factors.

Developed and evaluated using the Design Science Research methodology, CRAM was implemented as a web-based decision-support artifact capable of supporting systematic and consistent pre-sales evaluation. The empirical findings demonstrate that the model effectively differentiates readiness levels, improves opportunity prioritization, and reduces inefficient engagement with low-readiness clients.

The study contributes theoretically by extending traditional qualification approaches toward software-specific readiness assessment and contributes practically through the delivery of a deployable decision-support system suitable for real-world industrial application.

7. Limitations

Several limitations should be acknowledged. First, the evaluation of CRAM was conducted within a single organizational context, which may limit the generalizability of the findings across different industries or organizational structures. Second, the weighting mechanism relied primarily on expert-informed judgment and practical experience rather than formal statistical optimization techniques. Third, the evaluation dataset remains moderate in size relative to broader industrial applications.

Despite these limitations, the study provides strong initial evidence supporting the feasibility and practical value of structured client readiness assessment in software development environments.

8. Future Work

Future research should focus on validating CRAM across multiple organizations and industries to improve generalizability. Additional studies may incorporate data-driven weighting techniques such as the Analytic Hierarchy Process (AHP), Delphi-based expert consensus methods, or machine learning approaches to enhance scoring accuracy and predictive capability.

Further work may also investigate the integration of CRAM with Customer Relationship Management (CRM) systems and predictive analytics platforms to support automated pre-sales intelligence and long-term project outcome forecasting.

References

1. Amin, A., & Shahzad, B. (2024). Improving requirements engineering practices for large-scale software systems. *Requirements Engineering*. <https://doi.org/10.1007/s00766-024-00425-2>
2. Barros, A., & Tam, C. (2024). Agile software development projects: Unveiling the human-related critical success factors. *Information and Software Technology*, 170, 107432. <https://doi.org/10.1016/j.infsof.2024.107432>
3. Boehm, B., & Turner, R. (2004). *Balancing agility and discipline: A guide for the perplexed*. Addison-Wesley.
4. Chen, X., et al. (2024). Improving requirements quality through structured validation approaches. *Computers, Materials & Continua*, 78(3). <https://doi.org/10.32604/cmc.2024.053830>
5. Dibble, B. (2011). *The Sales Developer: Practical Guide to Outsourcing Your Sales Development*. McGraw-Hill.
6. Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1), 75–105. <https://doi.org/10.2307/25148625>
7. Hoy, C., & Xu, L. (2023). Challenges in agile requirements engineering: A systematic literature review. *Information*, 14(6), 322. <https://doi.org/10.3390/info14060322>
8. Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3), 45–77. <https://doi.org/10.2753/MIS0742-1222240302>
9. Qayyum, A., et al. (2024). Human factors influencing agile software development performance. *Frontiers in Computer Science*, 6, 1400750. <https://doi.org/10.3389/fcomp.2024.1400750>

10. Rackham, N. (1988). *SPIN Selling*. McGraw-Hill.
11. Schwaber, K., & Beedle, M. (2002). *Agile software development with Scrum*. Prentice Hall.
12. Standish Group. (2020). *CHAOS Report 2020: Beyond Infinity*. The Standish Group International.