Scientific Journal of Informatics
Vol. 13, No. 3, August 2026

p-ISSN 2407-7658 http://journal.unnes.ac.id/nju/index.php/sji e-ISSN 2460-0040

Improving Scrum Implementation Quality in Software Development through an Integrated Capability Maturity Assessment and Decision-Making Framework

Farand Farhansyah[1], Eko K. Budiardjo[2], Alex Ferdinansyah[3]
[1, 2, 3] Master of Information Technology, Faculty of Computing Sciences,
Universitas Indonesia, Indonesia

Abstract.
Purpose: This study evaluates the quality of Scrum implementation in the New Corporate Website project at XYZ Bank and proposes process improvement recommendations using an integrated framework based on the Capability Maturity Model Integration (CMMI) 2.0. CMMI 2.0 was selected because it provides comprehensive Practice Areas for assessing organizational capability beyond Scrum-specific maturity models.
Methods: A convergent parallel mixed-method case study was employed involving 14 AHP experts, 17 Scrum appraisal respondents, and 3 interview participants. Quantitative and qualitative data were collected through questionnaires, semi-structured interviews, and document observations. The Analytical Hierarchy Process (AHP) was used to prioritize organizational problems, the Weighted Scoring Model (WSM) mapped the prioritized problems to the most relevant CMMI 2.0 Practice Areas, and Scrum capability was evaluated using the Key Process Area (KPA) Rating and SCAMPI C appraisal approach.
Results: Five Practice Areas were selected for appraisal. Verification and Validation (VV) was the only Practice Area that achieved Capability Level 1 with a KPA score of 96.16%, whereas Project Planning (PLAN), Process and Quality Assurance (PQA), Peer Review (PR), and Requirements Development and Management (RDM) remained at Capability Level 0. Based on the identified weaknesses, 11 improvement recommendations were formulated and organized into short-, medium-, and long-term implementation phases to support continuous software process improvement.
Novelty: This study proposes an integrated framework that combines AHP, WSM, and CMMI 2.0 to systematically prioritize Practice Areas before capability assessment. The proposed framework provides a structured decision-making mechanism that links organizational problems with capability appraisal and offers practical guidance for improving Scrum implementation and supporting continuous software process improvement.

Keywords: CMMI 2.0, Capability Level, Scrum, Appraisal, Improvement Recommendations
Received July 2026 / **Revised** July 2026 / **Accepted** August 2026

INTRODUCTION

Digital transformation has become a strategic priority for organizations seeking to enhance competitiveness, improve operational efficiency, and sustain long-term business performance in increasingly dynamic and technology-driven environments. In Indonesia, the digital economy has experienced remarkable growth, increasing from USD 41 billion in 2019 to USD 77 billion in 2022 and projected to reach USD 130 billion by 2025 [1]. This growth has been driven primarily by the rapid expansion of e-commerce, digital financial services, transportation, online travel, and digital media [1]. Consequently, organizations are under increasing pressure to accelerate digital transformation while ensuring that software development processes remain adaptive, efficient, and capable of delivering high-quality digital products. Beyond adopting digital technologies, digital transformation requires organizations to continuously improve their internal processes, governance, and organizational capabilities to respond effectively to evolving business requirements [2].

As one of the leading banking institutions in Indonesia, XYZ Bank has recognized digital transformation as a strategic initiative to strengthen its competitiveness and enhance customer-oriented financial services.

*Corresponding author.
Email addresses: farand.farhansyah@ui.ac.id (Farhansyah), eko@cs.ui.ac.id (Budiardjo), alex.ferdinansyah@cs.ui.ac.id (Ferdinansyah)
DOI: https://doi.org/10.15294/sji.v13i3.58224

To support this initiative, the organization developed the New Corporate Website project using the Scrum framework. Scrum is one of the most widely adopted Agile software development methodologies because it emphasizes iterative development, continuous stakeholder collaboration, rapid feedback, and adaptability to changing requirements [3]. Previous studies have demonstrated that Scrum implementation can improve team productivity, communication, software quality, and project delivery when Agile practices are consistently applied [4], [5]. Nevertheless, these benefits largely depend on the maturity and capability of the underlying software development processes.

Despite adopting Scrum, the implementation of the New Corporate Website project at XYZ Bank still encounters several organizational and process-related challenges. Preliminary observations identified problems related to process management, people, technology, and organizational governance [6]. Figure 1 illustrates the Fishbone Diagram summarizing these root causes.

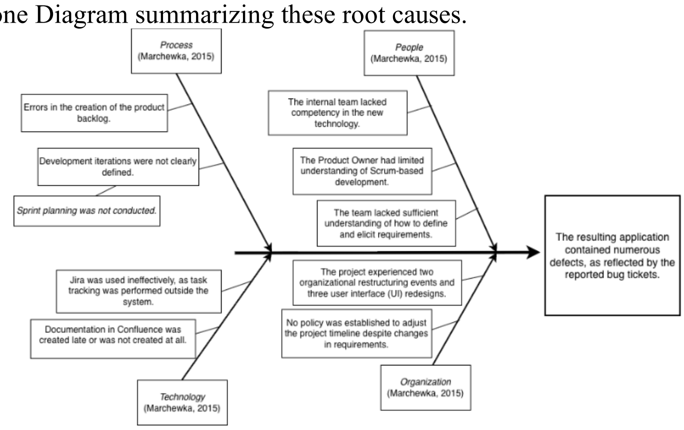

Figure 1. Fishbone Diagram

As illustrated in Figure 1, the identified problems include incomplete Product Backlog documentation, the absence of formal Sprint Planning practices, limited Scrum knowledge among Product Owners, software requirements dispersed across multiple repositories without a single source of truth, ineffective utilization of Jira, delayed documentation in Confluence, organizational restructuring, and the absence of policies for adjusting project timelines when requirements change. These issues indicate that Scrum implementation has not yet achieved the expected level of process capability and organizational maturity, potentially reducing software quality, development efficiency, and project predictability. Therefore, a structured capability assessment is required not only to evaluate the current implementation of Scrum but also to systematically identify organizational weaknesses and determine priority areas for process improvement.
Software process capability assessment has been widely adopted to evaluate Agile software development and identify opportunities for process improvement. Previous studies have employed maturity models such as the Scrum Maturity Model (SMM), Agile Maturity Model (AMM), and Capability Maturity Model Integration (CMMI). Abimaulana et al. [4] demonstrated the effectiveness of SMM and AMM in assessing Scrum implementation, while Wibawa et al. [7] applied CMMI-DEV Version 1.3 to improve requirements engineering processes. Similarly, Zelfia et al. [8] showed that Scrum maturity assessment can identify weaknesses in Agile implementation, and recent studies have confirmed that Scrum improves collaboration, communication, and software development efficiency [9]. These studies demonstrate the value of maturity assessment in supporting software process improvement.

However, existing studies primarily evaluate predefined maturity dimensions or Practice Areas without systematically determining which organizational problems should be prioritized before capability assessment. Moreover, the application of Capability Maturity Model Integration (CMMI) 2.0 to Scrum implementation remains limited, and only a few studies have integrated multi-criteria decision-making methods to support the selection of relevant Practice Areas. Consequently, process improvement initiatives may not always focus on the most critical organizational issues, highlighting the need for a more systematic and evidence-based evaluation approach.

To address these gaps, this study proposes an integrated framework combining the Fishbone Diagram, the Analytical Hierarchy Process (AHP), the Weighted Scoring Model (WSM), Capability Maturity Model Integration (CMMI) 2.0, and SCAMPI C. The Fishbone Diagram is used to identify organizational root causes, AHP prioritizes these problems, WSM determines the most relevant CMMI 2.0 Practice Areas, and

SCAMPI C evaluates their capability levels to formulate prioritized improvement recommendations. This integrated framework contributes theoretically by extending the application of CMMI 2.0 to Scrum capability assessment, methodologically by combining root-cause analysis, decision-making, and capability appraisal into a unified process, and practically by providing prioritized recommendations to support continuous software process improvement. Based on these contributions, this study addresses the following research questions: RQ1: What is the capability level of Scrum implementation in the New Corporate Website project at XYZ Bank based on CMMI 2.0? RQ2: Which CMMI 2.0 Practice Areas should be prioritized through the integration of the Fishbone Diagram, AHP, and WSM? RQ3: What improvement recommendations can enhance Scrum implementation quality? Accordingly, this study aims to evaluate the capability level of Scrum implementation, identify and prioritize organizational process weaknesses, and propose improvement recommendations that support continuous software process improvement and digital transformation initiatives.

METHODS

This study employed a convergent parallel mixed-method case study design [10] to evaluate Scrum implementation in the New Corporate Website project at XYZ Bank. This design was selected because quantitative and qualitative data were collected concurrently, analyzed independently, and integrated during the interpretation stage to provide a comprehensive assessment of Scrum implementation. Quantitative data were obtained through the Analytic Hierarchy Process (AHP) and Capability Maturity Model Integration (CMMI) 2.0 appraisal, while qualitative data were collected through semi-structured interviews and document observations. Integrating these data sources enabled methodological triangulation and improved the validity of the findings.

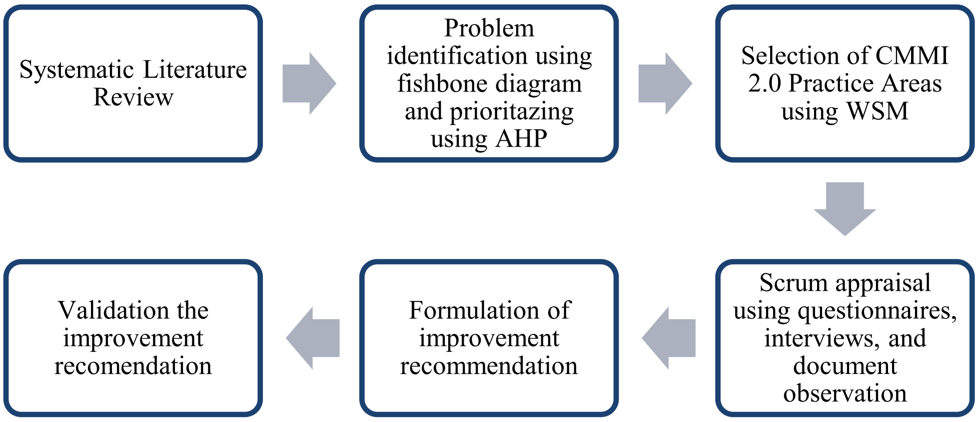

Figure 2. Research methodology

Figure 2 illustrates the proposed research framework, which consists of six stages: (1) Systematic Literature Review (SLR), (2) organizational problem identification and prioritization using the Fishbone Diagram and AHP, (3) selection of CMMI 2.0 Practice Areas using the Weighted Scoring Model (WSM), (4) Scrum capability appraisal using CMMI 2.0 and SCAMPI C, (5) formulation of improvement recommendations, and (6) expert validation. This sequential workflow ensures that organizational problems are systematically identified, prioritized, assessed, and translated into actionable process improvement recommendations.

The SLR was conducted using IEEE Xplore, ScienceDirect, and Taylor & Francis to identify relevant studies on Scrum evaluation and software process improvement. Following predefined inclusion and exclusion criteria, 473 publications were initially identified, reduced to 17 after title and abstract screening, and finally narrowed to five primary studies through full-text assessment [4], [7], [8], [11], [12]. These studies provided the theoretical foundation for selecting the research methods and identifying existing research gaps.

Organizational problems were identified through interviews with Scrum practitioners and analysis of project artifacts, including Product Backlog, Jira tickets, Confluence documentation, website analytics, and bug reports. The identified issues were categorized into four dimensions process, people, technology, and organization using the Fishbone Diagram. This classification served as the basis for prioritizing organizational problems before capability assessment.

The identified organizational problems were prioritized using the Analytical Hierarchy Process (AHP) with pairwise comparison judgments from 14 experts directly involved in Scrum implementation. Respondents were selected through purposive sampling based on their project involvement, Scrum experience, and understanding of software development processes. Pairwise comparisons followed Saaty's nine-point scale [13], and the priority weights were calculated using the normalized principal eigenvector. The consistency of the judgments was evaluated using the Consistency Ratio (CR), where a value of CR ≤ 0.10 indicates acceptable consistency [14]. The resulting global priority values were then used as input for the next stage. The Weighted Scoring Model (WSM) was employed to determine the most relevant CMMI 2.0 Practice Areas because it provides a simple and transparent mechanism for aggregating weighted organizational problems into alternative Practice Areas [15]–[17]. Compared with methods such as TOPSIS or VIKOR, WSM is more appropriate for additive scoring where each prioritized problem contributes proportionally to the final evaluation score. The WSM score was calculated as [18]–[20]:

$$\text{WSM} = \sum_{i=1}^{n} w_i x_i$$

Where w_i represents the global priority weight obtained from AHP and x_i indicates the relationship between an organizational problem and a CMMI 2.0 Common Problem. Based on the aggregated scores, five Practice Areas were selected for appraisal: Project Planning (PLAN), Process and Quality Assurance (PQA), Peer Review (PR), Requirements Development and Management (RDM), and Verification and Validation (VV).

The selected Practice Areas were evaluated using the CMMI 2.0 appraisal framework with the SCAMPI C approach. SCAMPI C was chosen because it provides a flexible diagnostic appraisal suitable for identifying process improvement opportunities without requiring the extensive evidence demanded by formal benchmark appraisals [21], [22]. Data were collected through questionnaires, semi-structured interviews, and document observations involving Jira, Confluence, Bitbucket, SonarQube, and Jenkins to ensure methodological triangulation. Questionnaire items and interview protocols were derived from the sub-practices defined in the selected Practice Areas. The appraisal results were quantified using the Key Process Area (KPA) Rating, which measures the degree of achievement of each Practice Area based on evidence collected from multiple data sources. The resulting KPA scores were subsequently interpreted to determine the corresponding Capability Level in accordance with the staged representation of CMMI 2.0, where all Level 1 sub-practices must be fully achieved before the organization can progress to the next capability level.

Based on the appraisal results, improvement recommendations were formulated for each Practice Area by addressing the identified process weaknesses. The recommendations were classified into three categories: New, referring to practices that need to be established; Enhance, referring to existing practices that require improvement; and Formalize, referring to informal practices that should be standardized and institutionalized. To facilitate implementation, the recommendations were organized into a non-overlapping roadmap consisting of short-term (0–3 months), medium-term (4–6 months), and long-term (7–9 months) initiatives.

The proposed recommendations were validated through expert judgment involving two domain experts: an external expert, a Senior Software Engineer from a leading Indonesian technology company, and an internal expert, the Department Head responsible for software development at XYZ Bank. The experts evaluated the relevance, feasibility, and applicability of each recommendation within the organizational context. Their feedback was incorporated to refine the final improvement roadmap.

This study involved respondents who were directly engaged in the New Corporate Website project and actively participated in Scrum activities. Fourteen experts participated in the AHP pairwise comparison process, while the Scrum capability appraisal involved 17 respondents, comprising 16 software developers and one Product Owner. In addition, three key informants the Scrum Master, the Frontend Team Lead, and the Solution Analyst Team Lead participated in semi-structured interviews to provide deeper insights into the appraisal results. Respondents were selected using purposive sampling because they possessed the knowledge and experience required to evaluate Scrum implementation. The respondents had practical experience in Agile software development and were actively involved in Scrum ceremonies, planning, development, testing, or project coordination.

To enhance the credibility of the findings, this study integrated quantitative and qualitative data through methodological triangulation. Quantitative results from the AHP analysis and CMMI 2.0 appraisal were interpreted alongside qualitative evidence obtained from interviews and project documentation. Each sub-practice was evaluated using at least two independent sources of evidence, ensuring consistency between questionnaire responses, interview findings, and document observations. This triangulation strengthened the validity and reliability of the capability assessment and the resulting improvement recommendations.

RESULT AND DISCUSSION

Literature Review Findings

To identify existing methods and research gaps in Scrum evaluation and software process improvement, a Systematic Literature Review (SLR) was conducted across three scientific databases: IEEE Xplore, ScienceDirect, and Taylor & Francis. The selection process involved title and abstract screening, followed by full-text review, to identify the most relevant studies. The results of the document selection process are presented in Table 1.

Table 1. Document screening from the SLR phase

Data Basis	Initiation Phase	Phase 1	Phase 2
IEEE Xplore	117	8	5
ScienceDirect	286	7	-
Taylor & Francis	70	2	-
Total	473	17	5

Table 1 presents the document selection process conducted during the Systematic Literature Review (SLR). A total of 473 documents were retrieved from IEEE Xplore, ScienceDirect, and Taylor & Francis databases. After title and abstract screening, 17 studies remained, and five primary studies were finally selected through full-text review. The selected studies mainly focus on Scrum maturity assessment, Scrum best practices, and software process improvement. The SLR findings indicate that previous studies generally evaluated Scrum implementation using predefined maturity models or focused on specific software development processes. However, limited studies have integrated organizational problem prioritization with CMMI 2.0-based Scrum appraisal. This limitation motivates the present study to propose an integrated approach combining AHP, WSM, and CMMI 2.0 for Scrum evaluation.

The SLR was performed systematically to ensure that only relevant and high-quality studies were selected. The overall literature review procedure adopted in this research is illustrated in Figure *3*.

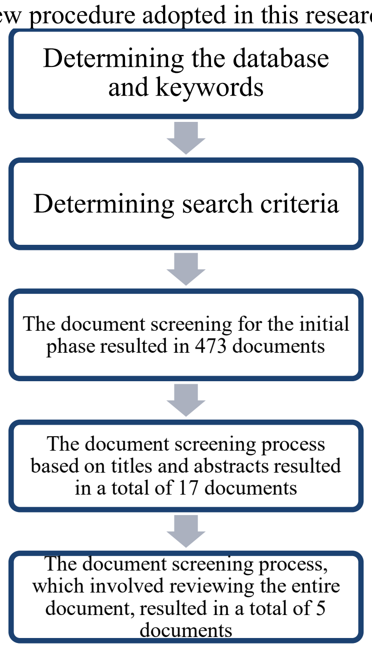

Figure 3 Overall SLR Process

Figure 3. illustrates the overall SLR process adopted in this study. The literature review was conducted systematically through three stages: initial search, title and abstract screening, and full-text

screening. This approach ensured that only relevant and high-quality studies were selected as references. The selected literature provides a comprehensive understanding of Scrum maturity assessment and software process improvement while highlighting the research gap addressed in this study. Therefore, the SLR serves not only as a theoretical foundation but also as a basis for identifying the novelty of the proposed research approach.

How is the capability level of Scrum implementation at XYZ Bank?

To determine the most critical issues affecting Scrum implementation, organizational problems were prioritized using AHP. The prioritization results, including criteria weights, sub criteria weights, and global priorities, are presented in Table 2.

Table 2. Ranking result from AHP Process

Criteria	Criteria's Weight	Subcriteria	Subcretia's Weight	*Global priority*	Rank
Process	30.31%	Errors in creating the Product Backlog	44.99%	13.64%	1
		Iteration is not defined in development	29.35%	8.90%	9
		Sprint Planning was not held	25.66%	7.78%	10
People	24.49%	The internal team lacks expertise in the new technology	27.32%	6.69%	12
		The PO lacks a good understanding of development using Scrum	31.83%	7.80%	11
		The team lacks a clear understanding of how to define and identify requirements	40.85%	10.00%	5
Technology	22.03%	Using Jira is ineffective when tasks are tracked outside the system	40.52%	8.93%	8
		Documentation on Confluence is created late or not at all	59.48%	13.10%	2
Organization	23.17%	There have been 2 organizational restructuring changes and 3 UI changes	52.68%	12.21%	3
		There is no policy regarding timeline adjustments, even if there are changes to the requirements	47.32%	10.96%	4

Table 2 presents the prioritization results of organizational problems using AHP. The analysis shows that the process dimension obtained the highest weight (30.31%), followed by people (24.49%), organization (23.17%), and technology (22.03%). These results indicate that process-related issues are the most critical obstacles to Scrum implementation at XYZ Bank. Furthermore, errors in Product Backlog creation emerged as the highest-priority problem, followed by delayed documentation in Confluence and organizational restructuring. The findings imply that improving process governance and requirements management should be the primary focus of organizational improvement efforts. Similar findings were reported by Wibawa et al., who stated that deficiencies in requirements engineering significantly affect software quality and development productivity.

The AHP-derived global priorities were then used as inputs for the Weighted Scoring Model (WSM) to map each organizational problem into CMMI 2.0 Common Problems: Poor Quality, Late Delivery, Inexperienced Personnel, etc. The WSM results are presented in Table 3.

Table 3. WSM Common Problem Mapping Results

Common Problems	Problems										Total Weight
	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	
Poor quality	1	1	1	1	1	1	1	1	1	1	1.00
Late delivery	1	1	1	1	1	0	1	1	1	1	0.87
Inexperienced personnel	1	0	1	1	1	1	0	1	1	1	0.77

Table 3 presents the WSM mapping results. 'Poor Quality' obtained the highest cumulative score of 1.00, encompassing all ten organizational problems, confirming it as the dominant Common Problem. 'Late Delivery' followed with 0.87 and 'Inexperienced Personnel' with 0.77. Based on these results, the five Practice Areas most associated with Poor Quality were selected for appraisal: Project Planning (PLAN), Process and Quality Assurance (PQA), Peer Review (PR), Requirements Development and Management (RDM), and Verification and Validation (VV), supplemented by Governance (GOV) and Implementation Infrastructure (II) as foundational Practice Areas. In CMMI 2.0, GOV and II are mandatory Practice Areas that must be included alongside any selected Practice Areas when computing Capability Levels, as they establish the foundational governance structures and implementation infrastructure upon which capability assessment depends [23].

After mapping organizational problems to CMMI 2.0 Common Problems and selecting the relevant Practice Areas, Scrum appraisal was conducted using the CMMI 2.0 framework and SCAMPI C approach. Data triangulation was applied by combining questionnaire responses, interview insights, and document observations, with a minimum of two independent evidence sources required per sub-practice. The Final KPA for each Practice Area was computed as the arithmetic mean of questionnaire-based and interview-based KPA scores. The capability profile of Scrum implementation is summarized in Table 4.

Table 4. Capability Profile of Scrum implementation

Practice Area	Level	KPA (%)	Interpretation	Capability Level
Project Planning	1	78.39	Largely Achieved	CL 0
Process and Quality Assurance	1	67.19	Largely Achieved	CL 0
Peer Review	1	72.74	Largely Achieved	CL 0
Requirements Development and Management	1	78.82	Largely Achieved	CL 0
Verification and Validation	1 2	96.16 68.03	Fully Achieved Largely Achieved	CL 1
Rate			78.66%	

Table 4 summarizes the capability profile of Scrum implementation at XYZ Bank. The five selected Practice Areas produced KPA scores ranging from 67.19% to 96.16%. Verification and Validation achieved Capability Level 1 with a KPA score of 96.16%, whereas the remaining Practice Areas were classified as Largely Achieved and therefore remained at Capability Level 0. The 68.03% shown in the final row represents the overall Capability Level 2 assessment, which was calculated separately according to the CMMI 2.0 appraisal procedure and should not be interpreted as the arithmetic mean of the five Practice Area scores.

Among the five selected Practice Areas, only Verification and Validation (VV) successfully achieved Capability Level 1 with a score of 96.16%, whereas Project Planning (PLAN), Process and Quality Assurance (PQA), Peer Review (PR), and Requirements Development and Management (RDM) remained at Capability Level 0. These results suggest that although verification and testing activities have been implemented effectively, the organization still faces challenges in planning, quality assurance, peer review, and requirements management. Similar findings were reported by Abimaulana et al. [4]and Zelfia et al.[8], who emphasized that inconsistencies in Scrum practices and insufficient requirements management negatively affect software process maturity. Therefore, improving these Practice Areas is essential to increase the overall capability level and enhance the effectiveness of Scrum implementation [24].

To further investigate the causes of low capability achievement, the weaknesses identified in each Practice Area are summarized in Table 5.

Table 5. Weaknesses by Practice Area

Code	Practice Area	Weakness Description	Sub-Practice	Severity
W-PLAN-01	PLAN	The Product Backlog is not fully documented by the Product Owner before Sprint Planning begins, as user stories lack acceptance criteria and a defined ready state, leading to incomplete sprint commitments.	1.1 PLAN, 2.7 PLAN	High
W-PLAN-02	PLAN	Story point estimates are never formally documented in any tool, causing sprint delays to recur and overtime to become normalized rather than addressing the root cause of poor estimation.	2.3 PLAN	High
W-PLAN-03	PLAN	The sprint scope is not bounded by team capacity because the Product Owner treats all requirements as equally urgent, resulting in overcommitted sprints and recurring delivery failures [7], [25], [26].	2.6 PLAN	High

Code	Practice Area	Weakness Description	Sub-Practice	Severity
W-PQA-01	PQA	New developers receive only informal verbal QA explanations with no documented onboarding material, creating inconsistent quality practices that depend on individual memory rather than institutionalized knowledge [27]–[29].	2.1 PQA, 2.2 GOV	Medium
W-PQA-02	PQA	Implement velocity-based sprint capacity planning in Jira and enforce a scope freeze policy that prohibits adding new requirements within 24 hours of Sprint Planning closure [25], [30], [31].	2.2 PQA, 2.2 II	Medium
W-PR-01	PR	Develop a structured QA onboarding program documented in Confluence, covering quality standards, code review expectations, testing procedures, and tool usage for all new developers [12], [16], [32].	2.3 PR	High
W-PR-02	PR	Maintain a dedicated Confluence page for coding standards with concrete code examples and a quarterly review cycle to ensure content remains current and trusted by the team [5], [8].	2.1 PR	Medium
W-RDM-01	RDM	Implement mandatory branch protection rules that technically prevent any code merge without at least one approved peer review, eliminating the possibility of deadline-driven bypass [4], [7].	1.1 RDM, 2.2 RDM	High
W-RDM-02	RDM	Introduce monthly knowledge-sharing sessions on code review best practices facilitated by senior developers, using real anonymized codebase examples to build review competency in junior team members [11], [32].	2.5 RDM	High
W-VV-01	VV	Mandate a Definition of Ready (DoR) checklist — including acceptance criteria, edge case documentation, and stakeholder sign-off — for every User Story before it can enter a sprint [7], [33], [34].	2.2 GOV, 2.1 II	Medium
W-VV-02	VV	Verification activities are inconsistently performed because test cases and verification evidence are not systematically documented or reviewed before deployment, reducing the effectiveness of defect detection and validation of software functionality [35], [36].	2.2 VV, 2.2 II	Medium

Table 5 summarizes the major weaknesses identified during the Scrum capability appraisal. Most high-severity weaknesses were observed in Project Planning (PLAN), Peer Review (PR), and Requirements Development and Management (RDM), indicating deficiencies in sprint planning, requirements management, and review practices that directly hinder the achievement of Capability Level 1. In contrast, the weaknesses identified in Process and Quality Assurance (PQA) and Verification and Validation (VV) were primarily related to inconsistent documentation, onboarding, and verification practices. These findings suggest that process standardization and institutionalization remain the primary challenges in Scrum implementation at XYZ Bank, thereby providing the basis for formulating the improvement recommendations presented in the subsequent section.

The findings reveal that most weaknesses are associated with planning, documentation, requirements management, and quality assurance activities. For example, incomplete Product Backlog documentation and informal Sprint Planning activities indicate that planning processes have not been standardized. In addition, scattered requirements and delayed documentation reduce the effectiveness of collaboration and traceability during software development. These weaknesses explain why four Practice Areas remained at Capability Level 0. Consequently, improvement initiatives should focus on establishing standardized processes, improving documentation practices, and strengthening requirements engineering activities to support continuous process improvement.

What improvements are required to improve Scrum implementation?

Based on the appraisal results and identified weaknesses, several improvement recommendations were formulated. The summary of the recommendations is presented in Table 6.

Table 6. Summary of The Recommendations

Code	PA	Type	Recommendations Summary	Priority	Target
REC-PLAN-01	PLAN	Formalize	Establish a mandatory Definition of Ready (DoR) requiring documented acceptance criteria, story points, and a Ready status in	Very High	0–3 month

Code	PA	Type	Recommendations Summary	Priority	Target
			Jira for all user stories before they enter Sprint Planning.		
REC-PLAN-02	PLAN	Enhance	Formalize story point estimation using Planning Poker guidelines documented in Confluence and institutionalize retrospectives as a data-driven mechanism for diagnosing estimation gaps.	Very High	0–3 month
REC-PLAN-03	PLAN	Enhance	Implement velocity-based sprint capacity planning in Jira and enforce a scope freeze policy that prohibits adding new requirements within 24 hours of Sprint Planning closure.	Very High	0–3 month
REC-PQA-01	PQA	New	Develop a structured QA onboarding program documented in Confluence, covering quality standards, code review expectations, testing procedures, and tool usage for all new developers.	Medium	0–3 month
REC-PQA-02	PQA	Enhance	Maintain a dedicated Confluence page for coding standards with concrete code examples and a quarterly review cycle to ensure content remains current and trusted by the team.	Medium	0–3 month
REC-PR-01	PR	Formalize	Implement mandatory branch protection rules that technically prevent any code merge without at least one approved peer review, eliminating the possibility of deadline-driven bypass.	High	1–3 month
REC-PR-02	PR	New	Introduce monthly knowledge-sharing sessions on code review best practices facilitated by senior developers, using real anonymized codebase examples to build review competency in junior team members.	Medium	1–3 month
REC-RDM-01	RDM	Formalize	Mandate a Definition of Ready (DoR) checklist — including acceptance criteria, edge case documentation, and stakeholder sign-off — for every User Story before it can enter a sprint.	High	0–3 month
REC-RDM-02	RDM	New	Consolidate all project requirements into Confluence as the Single Source of Truth (SSoT) with version history and cross-links to Jira and Figma, replacing the current fragmented multi-tool approach.	Very High	0–3 month
REC-VV-01	VV	New	Develop a formal VV training program documented in Confluence and delivered through hands-on workshops, teaching developers how to design effective test scenarios using the PTR Script process.	Medium	3–6 month
REC-VV-02	VV	New	Pilot automated regression testing in the CI/CD pipeline by automating a priority subset of existing PTR Script test cases to trigger automatically on each code push.	Medium	3–6 month

Table 6 summarizes the improvement recommendations proposed in this study. A total of eleven recommendations were formulated and classified into three categories: New, Enhance, and Formalize. The recommendations cover several important aspects, including Product Backlog standardization, improvement of Sprint Planning practices, centralized requirements management, enhancement of documentation processes, and formalization of quality assurance procedures. Compared with previous studies that primarily provide maturity assessments or best-practice recommendations, this study offers prioritized, context-aware recommendations derived from organizational problems and CMMI 2.0 appraisal results. Therefore, the recommendations are expected to provide practical guidance for improving Scrum implementation at XYZ Bank.

To facilitate the implementation of the proposed recommendations, a phased roadmap was developed. The implementation roadmap is shown in Figure *4*.

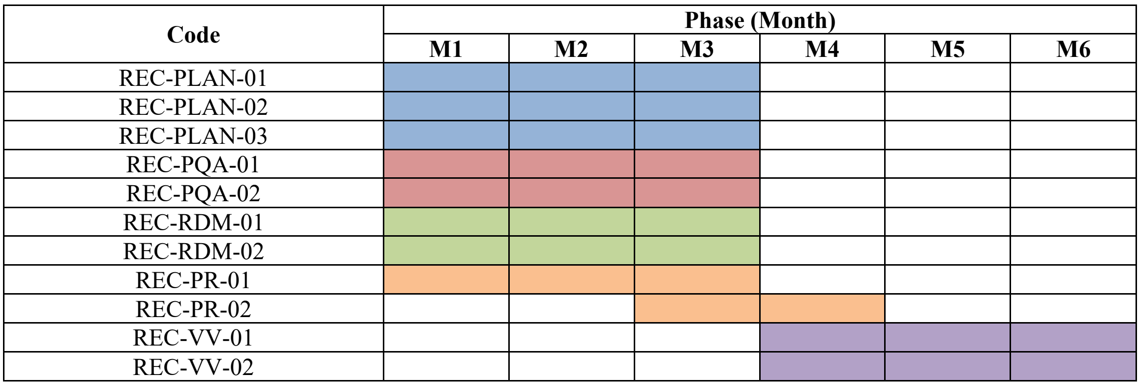

Figure 4. Implementation Roadmap of Improvement Recommendations Based on CMMI 2.0 Appraisal Results

Figure 4 presents the implementation roadmap of the proposed improvement recommendations based on the CMMI 2.0 appraisal results. High-priority initiatives related to Project Planning (PLAN), Process and Quality Assurance (PQA), Requirements Development and Management (RDM), and Peer Review governance (PR) are scheduled during the first three months as quick-win activities to address the most critical weaknesses identified in the appraisal. Peer Review capability development is implemented in the medium-term phase, while Verification and Validation (VV) improvement initiatives, including formal training and adoption of test automation, are planned for the later phase due to their higher resource, infrastructure, and organizational readiness requirements.

The proposed recommendations were validated by two domain experts: an external validator (a Senior Software Engineer at a leading Indonesian technology company) and an internal validator (a Department Head at XYZ Bank). Both validators confirmed the recommendations' relevance and feasibility, affirming that the proposed improvements are actionable within the organization's existing infrastructure and toolset. The findings indicate that the major challenges encountered in the project were not solely technical but also stemmed from deficiencies in project planning, requirements management, stakeholder involvement, and process governance. This suggests that the successful implementation of Scrum depends not only on adopting Agile practices but also on the organization's ability to consistently apply supporting processes that promote collaboration, transparency, and continuous improvement. Similar observations have been reported in previous studies, which emphasize that successful digital product development requires structured processes and continuous improvement mechanisms to maintain alignment between organizational objectives and stakeholder needs.

4. CONCLUSION

An integrated framework combining the Fishbone Diagram, Analytical Hierarchy Process (AHP), Weighted Scoring Model (WSM), Capability Maturity Model Integration (CMMI) 2.0, and SCAMPI C was employed to evaluate Scrum implementation in the New Corporate Website project at XYZ Bank. The results indicate that Verification and Validation (VV) was the only Practice Area to achieve Capability Level 1, whereas Project Planning (PLAN), Process and Quality Assurance (PQA), Peer Review (PR), and Requirements Development and Management (RDM) remained below the required capability level due to weaknesses in planning, process standardization, peer review, and requirements traceability. Based on these findings, eleven prioritized improvement recommendations and a phased implementation roadmap were developed to support continuous software process improvement. The proposed framework extends the application of CMMI 2.0 to Scrum capability assessment by integrating root-cause analysis, multi-criteria decision-making, and capability appraisal within a single evaluation process. From a practical perspective, the framework enables organizations to prioritize improvement initiatives according to organizational problems rather than predefined Practice Areas, thereby supporting more effective resource allocation and continuous improvement of Agile software development processes. Several limitations should be acknowledged. The findings are based on a single case study conducted within one banking organization and therefore may not be directly generalizable to other organizational contexts. In addition, the appraisal employed the SCAMPI C approach, which is intended for diagnostic assessment rather than formal CMMI certification, while part of the evidence relied on questionnaires and interviews that may introduce respondent bias despite the use of methodological triangulation. Future research could validate the proposed framework across different industries, conduct longitudinal evaluations to measure post-implementation improvements, apply more rigorous appraisal methods such as SCAMPI A or SCAMPI B, and expand the assessment to additional CMMI 2.0 Practice Areas, including Risk and Opportunity Management (RSK) and Estimation (EST), to provide a more comprehensive organizational capability profile.

REFERENCES

[1] Direktorat Jenderal Perbendaharaan (DJPb) Kementerian Keuangan RI, "Transformasi Digital untuk Masa Depan Ekonomi dan Bisnis di Indonesia," Direktorat Jenderal Perbendaharaan (DJPb) Kementerian Keuangan RI." .

[2] F. Abuhasan dan M. Moreb, "The Impact of the Digital Transformation on Customer Experience in Palestine Banks," *2021 International Conference on Information Technology (ICIT)*. hal. 43–48, 2021, doi: 10.1109/ICIT52682.2021.9491744.

[3] P. Amajuoyi, L. B. Benjamin, dan K. B. Adeusi, "Agile methodologies: Adapting product management to rapidly changing market conditions," *GSC Adv. Res. Rev.*, vol. 19, no. 2, hal. 249–267, 2024.

[4] K. C. Abimaulana, E. K. Budiardjo, K. Mahatma, dan A. Hidayati, "Evaluation of scrum-based software development process maturity using the SMM and AMM: A case of education technology startup," in *2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, 2021, hal. 1–5.

[5] S. S. Kurniawan, N. Z. Maharani, D. I. Sensuse, E. H. Purwaningsih, dan D. S. Hidayat, "Applying user centered design and system usability scale to design knowledge management system for exam proctors in higher education," *Sci. J. Informatics*, vol. 11, no. 4, hal. 1043–1056, 2024.

[6] N. Cucari, V. Lagasio, G. Lia, dan C. Torriero, "The impact of blockchain in banking processes: The Interbank Spunta case study," *Technol. Anal. Strateg. Manag.*, vol. 34, no. 2, hal. 138–150, 2022.

[7] A. S. Wibawa, E. K. Budiardjo, dan K. Mahatma, "Improving the quality of requirements engineering process in software development with agile methods: A case study telemedicine startup XYZ," in *2021 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS)*, 2021, hal. 1–5.

[8] H. Zelfia, T. Simanungkalit, dan T. Raharjo, "Comparison of scrum maturity between internal and external software development: a case study at one of the state-owned banks in Indonesia," in *2022 1st International Conference on Information System & Information Technology (ICISIT)*, 2022, hal. 312–317.

[9] L. Anggraeni, M. D. Haq, dan H. P. Pertiwi, "Development of School Exam Administration Based on LCDP with Scrum Method: An Innovation in Administrative Efficiency in Education," *Sci. J. Informatics*, vol. 11, no. 2, hal. 273–286, 2024.

[10] J. P. Takona, "Research design: qualitative, quantitative, and mixed methods approaches," *Qual. Quant.*, vol. 58, no. 1, hal. 1011–1013, 2024.

[11] A. Kurniawan, E. K. Budiardjo, dan K. Mahatma, "Scrum best practices recommendation: a media and community startup case study," in *2021 International Conference Advancement in Data Science, E-learning and Information Systems (ICADEIS)*, 2021, hal. 1–5.

[12] H. A. Muzakkiy dan Y. G. Sucahyo, "Evaluation of scrum framework implementation with scrum maturity model: a case study of PT XYZ, ABC Division," in *2023 International Conference on Computer Applications Technology (CCAT)*, 2023, hal. 41–46.

[13] A. Fauji, T. Raharjo, dan A. N. Fitriani, "Essential success drivers for banking legacy system migration using AHP.," *Tek. J. Sains dan Teknol.*, vol. 21, no. 2, hal. 337, 2025.

[14] X. Yun *et al.*, "A multi-dimensional AHP comprehensive evaluation of zero liquid discharge systems: A case study of chemical enterprise," *Environ. Res.*, hal. 124367, 2026.

[15] K. AL-Dosari dan N. Fetais, "Risk-management framework and information-security systems for small and medium enterprises (SMES): A meta-analysis approach," *Electronics*, vol. 12, no. 17, hal. 3629, 2023.

[16] M. Husni, "CMMI V2. 0 Maturity Level 2 and Scrum Applicability in Jordanian Agile Companies Based on Expert Review," 2024.

[17] Z. Sebestyén, J. Erdei, dan D. Alfreahat, "Impact of methodologies and standards on the owner's

economic benefit in projects," *Heliyon*, vol. 8, no. 2, 2022.
[18] A. V. Vitianingsih, S. S. K. Baharin, O. Othman, dan A. Suraji, "Empirical study of a spatial analysis for prone road traffic accident classification based on MCDM method," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 5, hal. 665–679, 2022.
[19] A. Lubis, N. Khairina, dan M. Riandra, "Comparison between Multiple Attribute Decision Making Methods through Objective Weighting Method in Determining Best Employee," *Int. J. Innov. Res. Comput. Sci. Technol.(IJIRCST)*, vol. 11, hal. 2347–5552, 2023.
[20] A. M. Alshamsi, H. El-Kassabi, M. A. Serhani, dan C. Bouhaddioui, "A multi-criteria decision-making (MCDM) approach for data-driven distance learning recommendations," *Educ. Inf. Technol.*, vol. 28, no. 8, hal. 10421–10458, 2023.
[21] R. Mildner, T. Ziller, dan F. Baiocchi, "Quality in the Automotive Industry—From Product to Process," in *Car IT Reloaded: Disruption in the Car Industry*, Springer, 2025, hal. 105–263.
[22] J. P. Castellanos Ardila, B. Gallina, dan F. Ul Muram, "Compliance checking of software processes: A systematic literature review," *J. Softw. Evol. Process*, vol. 34, no. 5, hal. e2440, 2022.
[23] M. Degerli, "Practical suggestions to successfully adopt the CMMI V2. 0 development for better process, performance, and products," in *2020 5th International Conference on Computer Science and Engineering (UBMK)*, 2020, hal. 126–129.
[24] M. Alshurideh, I. A. Akour, B. Al Kurdi, dan S. Hamadneh, "Exploring the impact of Scrum framework on project effectiveness: A quantitative analysis of agile implementation challenges and benefits," *Int. J. Theory Organ. Pract.*, vol. 4, no. 2, hal. 161–174, 2024.
[25] M. D. Kadenic, K. Koumaditis, dan L. Junker-Jensen, "Mastering scrum with a focus on team maturity and key components of scrum," *Inf. Softw. Technol.*, vol. 153, hal. 107079, 2023.
[26] W. Zayat dan O. Senvar, "Framework study for agile software development via scrum and Kanban," *Int. J. Innov. Technol. Manag.*, vol. 17, no. 04, hal. 2030002, 2020.
[27] D. Baxter dan N. Turner, "Why Scrum works in new product development: the role of social capital in managing complexity," *Prod. Plan. Control*, vol. 34, no. 13, hal. 1248–1260, 2023.
[28] E. L. Cano, J. M. García-Camús, J. Garzás, J. M. Moguerza, dan N. N. Sánchez, "A Scrum-based framework for new product development in the non-software industry," *J. Eng. Technol. Manag.*, vol. 61, hal. 101634, 2021.
[29] M. Mirzaei, V. J. Mabin, dan O. Zwikael, "Customising Hybrid project management methodologies," *Prod. Plan. Control*, vol. 36, no. 9, hal. 1188–1205, 2025.
[30] J. A. Hüllmann, K. Kimathi, dan P. Weritz, "Large-scale agile project management in safety-critical industries: A case study on challenges and solutions," *Inf. Syst. Manag.*, vol. 42, no. 2, hal. 138–160, 2025.
[31] T. L. Junker, A. B. Bakker, dan D. Derks, "Toward a theory of team resource mobilization: A systematic review and model of sustained agile team effectiveness," *Hum. Resour. Manag. Rev.*, vol. 35, no. 1, hal. 101043, 2025.
[32] O. A. Dada dan I. T. Sanusi, "The adoption of Software Engineering practices in a Scrum environment," *African J. Sci. Technol. Innov. Dev.*, vol. 14, no. 6, hal. 1429–1446, 2022.
[33] CMMI Institute, *CMMI Development V2.0.* CMMI Institute, 2018.
[34] S. Shafiee, Y. Wautelet, S. Poelmans, dan S. Heng, "An empirical evaluation of scrum training's suitability for the model-driven development of knowledge-intensive software systems," *Data Knowl. Eng.*, vol. 146, hal. 102195, 2023.
[35] M. K. Kushwaha, P. David, dan G. Suseela, "Automation and DevSecOps: streamlining security measures in financial system," in *2024 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, 2024, hal. 1–6.
[36] R. Rajab dan M. Alnoukari, "DevOps Integration With Capability Model Maturity Integration: A Systematic Mapping Review," *IEEE Access*, 2025.