

Theoretical and Applied Technological Science Review

Volume: 4 Issue: 2 Year: 2026

ISSN-2958-7824



Received: 2 May 2026

Revised: 20 June 2026

Accepted: 25 June 2026

DOI: <https://doi.org/10.64907/xkmf.v4i2.tatscr.1>

Research Article

From Coding to Collaboration: A Phenomenological Study of Communication Skill Development in Agile Learning Environments

Shafin Ahmed¹; Kazi Abdul Mannan²

¹ Department of Computer Science & Engineering (CSE),

² Department of Business Administration
Shanto-Mariam University of Creative Technology
Dhaka, Bangladesh

*Correspondence

Shafin Ahmed1

Email:

shafinahmed546@gmail.com

ABSTRACT

The increasing adoption of Agile methodologies in software development has transformed the competencies required of computer science graduates, emphasising communication and collaboration alongside technical expertise. This study explores how Agile learning environments facilitate the development of communication skills among learners transitioning from coding-focused tasks to collaborative practices. Grounded in social constructivism, experiential learning theory, and communities of practice, the research adopts a qualitative phenomenological approach using secondary data from peer-reviewed studies, case reports, and reflective accounts. The findings reveal that structured Agile practices, such as daily stand-ups, pair programming, and retrospectives, serve as critical mechanisms for fostering communication through iterative interaction, reflection, and feedback. These environments promote dialogic learning, enhance confidence, and support the development of professional communicative identities. However, challenges including unequal participation, communication anxiety, and cultural diversity persist. The study concludes that Agile learning environments provide a robust pedagogical framework for integrating communication skill development into technical education, offering valuable implications for curriculum design and instructional practice in software engineering education.

Keywords: Agile learning, communication skills, phenomenology, collaborative learning, software engineering education, Scrum, experiential learning

Copyright: 2026 by the authors. Licensee KMF Publishers (www.kmf-publishers.com). This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The rapid transformation of the global software industry has significantly reshaped the skill set required of computer science and software engineering graduates. Traditionally, computing education emphasised technical competencies such as programming, algorithmic thinking, and system design. While these remain foundational, the contemporary workplace increasingly demands a broader range of competencies, particularly communication, collaboration, and adaptability (Sommerville, 2016; Kitchenham et al., 2017). This shift is largely driven by the adoption of Agile methodologies, which prioritise iterative development, cross-functional teamwork, and continuous stakeholder engagement (Beck et al., 2001; Schwaber & Sutherland, 2020).

Agile methodologies have redefined software development as a highly collaborative and communicative process. Unlike traditional waterfall models, which rely on sequential phases and extensive documentation, Agile approaches emphasise frequent interaction, rapid feedback cycles, and adaptive planning (Rigby et al., 2016). Practices such as daily stand-up meetings, sprint reviews, retrospectives, and pair programming require developers to communicate regularly, articulate ideas clearly, and engage in constructive dialogue. Consequently, communication is no longer a peripheral skill but a central competency in Agile environments.

In response to these industry changes, educational institutions have begun integrating Agile frameworks into their curricula. Agile learning environments aim to simulate real-world development contexts, enabling students to experience collaborative workflows and develop essential soft skills alongside technical expertise (Mahnič, 2012). These environments often involve team-based projects, iterative development cycles, and role-based responsibilities, mirroring professional Agile teams. As a result, students are exposed to continuous communication practices that foster both interpersonal and professional growth.

Despite these advancements, a significant gap remains between traditional coding-focused education and the collaborative demands of Agile environments. Many students enter software engineering programs with strong technical skills but limited experience in teamwork and communication. The transition from individual coding tasks to collaborative problem-solving can be challenging, as it requires not only technical proficiency but also the ability to articulate ideas, negotiate solutions, and engage in reflective dialogue (Lethbridge et al., 2007). This transition is particularly complex in diverse educational settings, where differences in language, culture, and communication styles may further influence learning experiences.

Communication skills in software engineering encompass a wide range of competencies, including verbal communication, written documentation, active listening, conflict resolution, and feedback exchange (Pikkarainen et al., 2008).

In Agile environments, these skills are continuously practised and refined through structured interactions. For example, daily stand-ups require concise and focused communication, while retrospectives encourage reflective and critical discussions about team performance. Pair programming, on the other hand, facilitates real-time collaboration and mutual learning, enabling students to develop both technical and interpersonal communication skills simultaneously.

From a pedagogical perspective, the integration of Agile practices aligns with contemporary learning theories that emphasise active, experiential, and social learning. Agile learning environments provide opportunities for students to engage in authentic tasks, collaborate with peers, and reflect on their experiences, thereby fostering deeper learning and skill development (Kolb, 1984; Vygotsky, 1978). However, the effectiveness of these environments in developing communication skills remains an area of ongoing research, particularly from a phenomenological perspective that focuses on learners' lived experiences.

This study addresses this gap by exploring how Agile learning environments facilitate the development of communication skills among students transitioning from coding-centric to collaborative practices. Adopting a qualitative phenomenological approach based on secondary data, the study synthesises insights from existing literature, case studies, and reflective accounts. By examining the lived experiences of learners as documented in these sources, the study seeks to uncover the underlying processes

through which communication skills are developed in Agile contexts.

The central research question guiding this study is:

How do Agile learning environments facilitate the development of communication skills among learners transitioning from coding-focused tasks to collaborative practices?

To answer this question, the study aims to achieve the following objectives:

- To analyse the role of Agile practices in promoting communication among learners
- To identify key factors that influence communication skill development in Agile learning environments
- To explore the challenges and limitations associated with communication in collaborative settings
- To contribute to the theoretical and pedagogical understanding of Agile-based education

By addressing these objectives, this research contributes to the growing body of literature on software engineering education and Agile pedagogy. It highlights the importance of integrating communication-focused learning experiences into technical curricula and provides insights into how educators can design more effective and inclusive learning environments.

2. Literature Review

Agile learning environments represent a pedagogical shift toward iterative,

collaborative, and student-centred learning. Rooted in the principles of the Agile Manifesto (Beck et al., 2001), these environments emphasise flexibility, responsiveness to change, and continuous improvement. In educational contexts, Agile methodologies are often implemented through frameworks such as Scrum and Extreme Programming (XP), which structure learning activities around sprints, team roles, and regular feedback cycles (Schwaber & Sutherland, 2020).

Empirical studies have demonstrated that Agile-based education enhances student engagement and learning outcomes. For instance, Mahnič (2012) found that incorporating Scrum into software engineering courses improved students' understanding of project management and teamwork. Similarly, Pope-Ruark (2015) argued that Agile pedagogy fosters a culture of collaboration and accountability, encouraging students to take ownership of their learning.

Agile learning environments also support active learning by involving students in real-world problem-solving tasks. Rather than passively receiving information, students actively participate in the learning process, collaborating with peers and applying theoretical knowledge to practical scenarios (Prince, 2004). This approach aligns with constructivist learning theories, which emphasise the role of social interaction in knowledge construction.

2.1 Communication Skills in Software Engineering Education

Communication skills are widely recognised as essential for success in software engineering. These skills encompass both technical communication (e.g., writing documentation, explaining code) and interpersonal communication (e.g., teamwork, conflict resolution) (Kitchenham et al., 2017). Despite their importance, communication skills have historically been underemphasized in computing curricula, which tend to prioritise technical competencies (Lethbridge et al., 2007).

Research indicates that communication failures are a major contributor to software project challenges. Pikkarainen et al. (2008) found that ineffective communication can lead to misunderstandings, delays, and reduced productivity. Conversely, effective communication enhances team coordination, improves problem-solving, and facilitates knowledge sharing.

In educational settings, developing communication skills requires deliberate pedagogical strategies. Collaborative learning approaches, such as group projects and peer reviews, provide opportunities for students to practice communication in meaningful contexts (Johnson & Johnson, 2009). Agile learning environments extend these opportunities by embedding communication into daily practices and workflows.

2.2 Agile Practices as Mechanisms for Communication Development

Agile methodologies incorporate several practices that inherently promote

communication. Daily stand-up meetings, for example, require team members to share updates, discuss challenges, and coordinate tasks. These brief but frequent interactions help students develop concise and effective communication skills (Rigby et al., 2016).

Pair programming is another key practice that facilitates communication. By working in pairs, students engage in continuous dialogue, explaining their thought processes and providing feedback to their partners. Research has shown that pair programming not only improves code quality but also enhances communication and collaboration skills (Williams & Kessler, 2002).

Retrospectives provide a structured space for reflection and feedback. During these sessions, team members discuss what went well, what could be improved, and how to enhance future performance. This practice encourages open communication, critical thinking, and mutual respect (Derby & Larsen, 2006).

Additionally, Agile roles such as Scrum Master and Product Owner require students to adopt different communication styles and responsibilities. These roles expose students to various aspects of communication, including facilitation, negotiation, and stakeholder engagement.

2.3 Theoretical Perspectives on Communication and Learning

The development of communication skills in Agile learning environments can be understood through several theoretical lenses. Social constructivism, as proposed by Vygotsky (1978), emphasises the role of social interaction in learning. According to

this theory, knowledge is constructed through dialogue and collaboration, making communication a central component of the learning process.

Experiential learning theory (Kolb, 1984) further highlights the importance of learning through experience and reflection. Agile practices provide iterative cycles of action and reflection, enabling students to continuously refine their communication skills.

The concept of communities of practice (Wenger, 1998) also offers valuable insights. In Agile teams, students participate in shared activities and develop a collective understanding of their work. Communication serves as the primary means through which knowledge is shared and constructed within these communities.

2.4 Challenges in Communication within Agile Learning Environments

While Agile learning environments offer significant benefits, they also present challenges. One common issue is unequal participation, where some students dominate discussions while others remain passive (Hoda et al., 2013). This imbalance can hinder the development of communication skills among less active participants.

Cultural and linguistic differences may also affect communication. In diverse classrooms, students may have varying levels of proficiency in the language of instruction, which can impact their ability to participate effectively (Robles, 2012). Additionally,

differences in communication styles may lead to misunderstandings or conflicts.

Another challenge is the lack of confidence among students, particularly those who are introverted or inexperienced in collaborative settings. These students may struggle to express their ideas or engage in discussions, limiting their opportunities for skill development.

Addressing these challenges requires intentional instructional design and supportive learning environments. Educators can implement strategies such as structured communication protocols, peer mentoring, and inclusive facilitation techniques to ensure equitable participation.

2.5 Research Gap

Although existing literature highlights the importance of communication in Agile environments, there is a need for a deeper exploration of how these skills are developed from a phenomenological perspective. Most studies focus on outcomes rather than the lived experiences of learners. This study addresses this gap by synthesising qualitative secondary data to provide a more nuanced understanding of communication skill development in Agile learning contexts.

3. Theoretical Framework

The present study is grounded in an integrative theoretical framework that combines social constructivism, experiential learning theory, and communities of practice to explain how communication skills develop within Agile learning environments. These theoretical perspectives collectively provide a robust lens for understanding the transition

from individual coding practices to collaborative, communication-intensive workflows.

3.1 Social Constructivism

Social constructivism, rooted in the work of Vygotsky (1978), posits that learning is inherently a social process mediated through interaction, language, and cultural tools. According to this perspective, knowledge is co-constructed through dialogue and collaboration rather than acquired individually. In Agile learning environments, communication is not merely a medium for exchanging information but a fundamental mechanism through which learning occurs.

Agile practices such as daily stand-ups, sprint reviews, and retrospectives exemplify social constructivist principles by creating structured opportunities for interaction. Through these practices, learners articulate their understanding, negotiate meanings, and co-construct solutions to complex problems. The concept of the Zone of Proximal Development (ZPD) is particularly relevant, as it highlights how learners can achieve higher levels of understanding through collaboration with more knowledgeable peers (Vygotsky, 1978). In pair programming and team-based tasks, students often operate within each other's ZPD, facilitating both cognitive and communicative development.

Furthermore, language plays a central role in social constructivism. In Agile environments, students are required to verbalise technical concepts, justify decisions, and engage in reflective discussions. These communicative acts contribute to deeper cognitive processing and knowledge internalisation (Mercer &

Littleton, 2007). Thus, communication skills are not only developed as a byproduct of collaboration but are integral to the learning process itself.

3.2 Experiential Learning Theory

Kolb's (1984) experiential learning theory provides a complementary perspective by emphasising learning as a cyclical process involving concrete experience, reflective observation, abstract conceptualisation, and active experimentation. Agile learning environments inherently align with this cycle through their iterative structure.

Each Agile sprint represents a complete learning cycle. Students engage in concrete experiences by working on coding tasks, participating in reflective observation during retrospectives, developing abstract conceptualisations by identifying patterns and strategies, and applying these insights through active experimentation in subsequent sprints. Communication is embedded throughout this cycle, particularly during the reflection and conceptualisation phases.

For example, retrospectives require students to discuss what worked well and what did not, fostering reflective communication. This process not only enhances self-awareness but also improves the ability to provide and receive constructive feedback. Similarly, sprint planning sessions involve collaborative decision-making, where students must articulate ideas, negotiate priorities, and reach consensus.

Experiential learning theory also underscores the importance of feedback in skill

development. Agile environments provide continuous feedback through peer interactions, instructor guidance, and iterative evaluations. This feedback loop enables learners to refine their communication skills over time, adapting their approaches based on experience (Kolb, 1984).

3.3 Communities of Practice

The concept of communities of practice (CoP), introduced by Wenger (1998), further enriches the theoretical framework by focusing on learning as participation in a shared social context. A community of practice is characterised by mutual engagement, joint enterprise, and shared repertoire. Agile teams in educational settings closely resemble such communities, where students collaborate toward common goals and develop shared practices.

Within these communities, communication serves as the primary means of participation and knowledge exchange. Newcomers gradually move from peripheral participation to full engagement, a process known as legitimate peripheral participation (Lave & Wenger, 1991). In Agile learning environments, novice learners initially observe and contribute minimally but gradually become more active participants as they gain confidence and competence.

Agile roles such as Scrum Master and Product Owner facilitate this process by distributing responsibilities and encouraging diverse forms of participation. Students learn to communicate in different contexts, including facilitation, coordination, and stakeholder interaction. Over time, they

develop a shared language and understanding of Agile practices, which enhances both team cohesion and communication effectiveness.

Communities of practice also emphasise identity formation. As students engage in Agile practices, they begin to see themselves as members of a professional community, adopting the communication norms and behaviours associated with software development (Wenger, 1998). This identity shift is crucial for the transition from academic learning to professional practice.

3.4 Integrative Perspective

By integrating these three theoretical perspectives, the study conceptualises communication skill development as a dynamic and socially situated process. Social constructivism explains how communication facilitates learning through interaction; experiential learning theory highlights the role of iterative practice and reflection; and communities of practice emphasise participation and identity formation.

Together, these frameworks provide a comprehensive understanding of how Agile learning environments support the development of communication skills. They underscore the importance of designing educational experiences that promote interaction, reflection, and collaboration, thereby preparing students for the communicative demands of modern software development.

4. Methodology

This study adopts a qualitative phenomenological research design to explore how communication skills are developed

within Agile learning environments. Phenomenology is concerned with understanding individuals' lived experiences and the meanings they ascribe to those experiences (Creswell & Poth, 2018; Moustakas, 1994). In this context, the study seeks to capture the essence of learners' experiences as they transition from coding-focused tasks to collaborative practices.

Unlike empirical phenomenological studies that rely on primary data such as interviews, this research employs a secondary data-based phenomenological approach. This involves analysing existing qualitative data sources, including published research articles, case studies, and reflective reports, to identify common themes and patterns. This approach is particularly suitable for synthesising a broad range of experiences across different contexts.

4.1 Data Sources and Selection Criteria

The study utilises multiple sources of secondary data to ensure richness and diversity of perspectives. These sources include:

- Peer-reviewed journal articles on Agile education and communication
- Conference proceedings in software engineering and education
- Case studies documenting Agile implementation in classrooms
- Reflective student reports and teaching narratives

To ensure the quality and relevance of data, the following inclusion criteria were applied:

- Studies published in reputable academic journals or conferences
- Research focusing on Agile methodologies in educational settings
- Studies addressing communication, collaboration, or soft skill development
- Qualitative or mixed-method studies providing descriptive insights

Exclusion criteria included studies focusing solely on technical outcomes without addressing communication or collaboration.

4.2 Data Analysis Procedure

The data were analysed using thematic analysis, following the framework proposed by Braun and Clarke (2006). This method is widely used in qualitative research for identifying, analysing, and reporting patterns within data.

The analysis involved the following steps:

Familiarisation with Data: The researcher engaged in repeated reading of selected sources to gain a comprehensive understanding of the content. Notes were taken to identify initial impressions and recurring ideas.

Coding: Relevant segments of text were coded based on their relation to communication skill development. Codes included categories such as “team interaction,” “feedback practices,” “communication challenges,” and “role-based communication.”

Theme Development: Codes were grouped into broader themes that captured significant aspects of the data. For example, codes related to stand-ups, pair programming, and

retrospectives were grouped under “structured communication practices.”

Theme Review and Refinement: Themes were reviewed to ensure coherence and consistency. Redundant or overlapping themes were merged, and clear definitions were established for each theme.

Interpretation: The final themes were interpreted in relation to the theoretical framework, linking empirical findings to social constructivism, experiential learning, and communities of practice.

4.3 Trustworthiness and Rigour

To ensure the rigour of the study, several strategies were employed:

- **Credibility:** Data were drawn from multiple reputable sources to ensure validity
- **Dependability:** A systematic and transparent analysis process was followed
- **Confirmability:** Interpretations were grounded in data and supported by existing literature
- **Transferability:** Findings are presented with sufficient detail to allow application in similar contexts

These criteria align with established standards for qualitative research (Lincoln & Guba, 1985).

4.4 Ethical Considerations

As the study relies on secondary data, it does not involve direct interaction with human participants. However, ethical considerations were maintained by:

- Properly citing all sources

- Avoiding misrepresentation of original findings (Mannan & Farhana, 2026)
- Respecting intellectual property rights

4.5 Limitations

The use of secondary data presents certain limitations. The study relies on previously published findings, which may not fully capture the diversity of learners' experiences. Additionally, the absence of primary data limits the ability to explore specific contexts in depth.

Despite these limitations, the study provides valuable insights by synthesising a wide range of perspectives and identifying common patterns in communication skill development.

5. Findings and Analysis

The thematic analysis of secondary qualitative data revealed that Agile learning environments significantly shape the development of communication skills through structured practices, collaborative engagement, and iterative reflection. The findings are organised into six major themes: structured communication rituals, dialogic learning through pair programming, reflective communication in retrospectives, role-based communicative identity formation, feedback-rich environments, and barriers and inequalities in communication development.

5.1 Structured Communication Rituals as Skill-Building Mechanisms

One of the most prominent findings is the role of structured Agile rituals, particularly daily stand-ups, sprint planning, and sprint reviews, in cultivating communication competencies. These rituals create predictable and repetitive opportunities for verbal interaction, thereby normalising communication as a core component of the learning process.

Daily stand-up meetings require participants to succinctly answer three key questions: what they have done, what they plan to do, and what obstacles they face. This format enforces clarity, brevity, and relevance in communication. Over time, students develop the ability to distil complex technical information into concise updates, a skill highly valued in professional contexts (Rigby et al., 2016). Furthermore, the regularity of these interactions reduces communication anxiety and builds confidence, particularly for students who initially struggle with public speaking.

Sprint planning sessions, on the other hand, involve collaborative decision-making and negotiation. Students must articulate their ideas, justify their choices, and reach consensus with team members. This process enhances persuasive communication and critical thinking skills. Similarly, sprint reviews require students to present their work to peers and instructors, fostering presentation skills and the ability to respond to feedback effectively.

These findings align with the work of Pikkarainen et al. (2008), who emphasised that Agile practices enhance communication frequency and transparency. In educational settings, these structured interactions function as scaffolds that guide students in developing effective communication habits.

5.2 Pair Programming as a Site of Dialogic Learning

Pair programming emerged as a powerful mechanism for communication skill development. In this practice, two students work collaboratively on a single task, alternating between the roles of “driver” and “navigator.” This arrangement necessitates continuous dialogue, as both participants must explain their reasoning, provide feedback, and coordinate their actions.

The data indicate that pair programming fosters dialogic learning, where knowledge is constructed through conversation and mutual engagement (Mercer & Littleton, 2007). Students reported increased confidence in explaining technical concepts and greater awareness of different problem-solving approaches. The need to verbalise thought processes enhances metacognitive skills, enabling learners to reflect on their own understanding.

Moreover, pair programming promotes active listening and empathy. Students must pay close attention to their partner’s ideas and perspectives, which fosters mutual respect and collaboration. This aligns with findings by Williams and Kessler (2002), who demonstrated that pair programming improves both technical outcomes and interpersonal skills.

However, the effectiveness of pair programming depends on the quality of interaction. In some cases, dominant participants may overshadow quieter peers, limiting opportunities for equitable communication. This highlights the importance of structured facilitation and role rotation to ensure balanced participation.

5.3 Reflective Communication in Retrospectives

Retrospectives play a crucial role in developing reflective communication skills. These sessions provide a structured space for teams to evaluate their performance, identify challenges, and propose improvements. Unlike other Agile practices that focus on task completion, retrospectives emphasise reflection and learning.

The analysis reveals that retrospectives encourage critical and constructive communication. Students learn to articulate their experiences, express concerns, and provide feedback in a respectful manner. This process enhances emotional intelligence and conflict resolution skills, which are essential for effective teamwork (Derby & Larsen, 2006).

Retrospectives also foster a culture of psychological safety, where students feel comfortable sharing their thoughts without fear of judgment. This environment is critical for open communication and honest feedback. Research by Edmondson (1999) suggests that psychological safety is a key determinant of team learning and performance.

Furthermore, retrospectives facilitate collective sense-making, where team

members collaboratively interpret their experiences and develop shared understandings. This aligns with social constructivist principles, emphasising the role of dialogue in knowledge construction (Vygotsky, 1978).

5.4 Role-Based Communication and Identity Formation

Agile learning environments assign specific roles to team members, such as Scrum Master, Product Owner, and Developer. These roles require distinct communication styles and responsibilities, providing students with diverse communicative experiences.

For instance, the Scrum Master acts as a facilitator, guiding discussions and ensuring effective communication within the team. This role develops leadership and facilitation skills. The Product Owner, on the other hand, is responsible for communicating requirements and priorities, which enhances negotiation and stakeholder communication skills.

The rotation of roles allows students to experience different perspectives and develop a broad range of communication competencies. Over time, students begin to internalise these roles, contributing to the formation of a professional identity. Wenger (1998) argues that identity formation is a central aspect of participation in communities of practice, and communication plays a key role in this process.

This role-based approach also highlights the contextual nature of communication. Students learn to adapt their communication style based on their role and audience, an essential skill in professional environments.

5.5 Feedback-Rich Environments and Iterative Communication

Agile learning environments are characterised by continuous feedback, which plays a critical role in communication skill development. Feedback occurs at multiple levels, including peer feedback, instructor feedback, and self-reflection.

The iterative nature of Agile practices ensures that feedback is timely and actionable. For example, during sprint reviews, students receive immediate feedback on their work, which they can incorporate into subsequent iterations. This process reinforces learning and promotes continuous improvement (Kolb, 1984).

Feedback also enhances communication by encouraging students to articulate their thoughts and respond to others' perspectives. The ability to give and receive feedback is a key component of effective communication, and Agile environments provide ample opportunities to practice this skill.

However, the quality of feedback is crucial. Constructive feedback that is specific, respectful, and actionable is more effective than vague or overly critical comments. Educators play a vital role in modelling and guiding effective feedback practices.

5.6 Barriers and Inequalities in Communication Development

Despite the numerous benefits, the analysis identified several barriers to communication skill development in Agile learning environments. One major challenge is unequal participation, where some students dominate discussions while others remain

passive. This imbalance can limit the opportunities for less active students to develop their communication skills (Hoda et al., 2013).

Cultural and linguistic diversity also influences communication. Students from different backgrounds may have varying communication styles and levels of language proficiency, which can affect their participation. For example, students who are less fluent in the language of instruction may struggle to express their ideas, leading to reduced engagement (Robles, 2012).

Additionally, communication anxiety and lack of confidence can hinder participation. Introverted students or those with limited prior experience in collaborative settings may find it challenging to engage in discussions. This highlights the need for supportive and inclusive learning environments that encourage all students to participate.

Finally, the effectiveness of Agile practices depends on proper implementation. Without adequate guidance, these practices may become superficial, reducing their impact on communication development.

6. Discussion

The findings of this study provide a comprehensive understanding of how Agile learning environments facilitate communication skill development, offering both theoretical and practical insights. By interpreting the findings through the lenses of social constructivism, experiential learning theory, and communities of practice, this section deepens the analysis and situates the

results within broader educational and professional contexts.

6.1 Communication as a Socially Constructed Competency

From a social constructivist perspective, communication emerges as both a means and an outcome of learning. The findings demonstrate that Agile practices create environments where knowledge is co-constructed through interaction. Daily stand-ups, pair programming, and retrospectives serve as platforms for dialogue, enabling students to share ideas, negotiate meanings, and build collective understanding (Vygotsky, 1978).

This aligns with Mercer and Littleton's (2007) concept of dialogic teaching, where learning occurs through structured conversation. In Agile environments, communication is not incidental but deliberately embedded in the learning process. This integration ensures that students continuously practice and refine their communication skills.

Moreover, the findings highlight the importance of language as a cognitive tool. By articulating their thoughts, students engage in deeper processing and reflection, which enhances both understanding and communication. This reinforces the idea that communication skills are inseparable from cognitive development.

6.2 Experiential Learning and Iterative Skill Development

The iterative nature of Agile practices aligns closely with Kolb's (1984) experiential learning cycle. Each sprint provides

opportunities for action, reflection, and improvement, creating a continuous learning loop. Communication skills are developed through repeated practice and feedback, enabling students to refine their abilities over time.

For example, the progression from initial participation in stand-ups to confident presentation in sprint reviews illustrates the cumulative effect of experiential learning. Students learn not only from their own experiences but also from observing and interacting with peers. This process fosters both individual and collective growth.

The role of feedback is particularly significant in this context. Continuous feedback allows students to identify areas for improvement and adjust their communication strategies accordingly. This iterative process is essential for developing complex skills such as communication, which cannot be mastered through theoretical instruction alone.

6.3 Communities of Practice and Identity Formation

The concept of communities of practice provides a valuable framework for understanding how communication skills are embedded in social participation. Agile teams function as learning communities where students engage in shared activities and develop a collective identity (Wenger, 1998).

The findings indicate that role-based participation plays a key role in this process. By assuming different roles, students gain exposure to various communication contexts and responsibilities. This diversity of experiences contributes to the development

of a professional identity, where students begin to see themselves as effective communicators within a team.

Legitimate peripheral participation (Lave & Wenger, 1991) is also evident in the findings. Novice learners initially participate at a peripheral level but gradually become more active contributors as they gain confidence and competence. This progression is facilitated by supportive team dynamics and structured Agile practices.

6.4 Addressing Challenges and Enhancing Inclusivity

While Agile learning environments offer significant benefits, the findings underscore the need to address challenges related to participation, diversity, and confidence. Unequal participation can hinder communication development, particularly for less active students. Educators must implement strategies to ensure equitable participation, such as structured turn-taking and role rotation.

Cultural and linguistic diversity presents both challenges and opportunities. While differences in communication styles may lead to misunderstandings, they also enrich the learning experience by exposing students to diverse perspectives. Educators should foster an inclusive environment that values diversity and supports language development.

Communication anxiety is another critical issue. Creating a psychologically safe environment is essential for encouraging participation. Techniques such as small group discussions, peer support, and positive

reinforcement can help build confidence and reduce anxiety.

6.5 Implications for Pedagogy and Practice

The findings have important implications for software engineering education. First, they highlight the need to integrate communication-focused activities into technical curricula. Agile practices provide an effective framework for achieving this integration.

Second, educators should emphasise the importance of communication as a core competency rather than a supplementary skill. This requires a shift in assessment practices to include communication-related outcomes.

Third, the study suggests that effective implementation of Agile practices is crucial for achieving desired learning outcomes. Educators must provide clear guidance and support to ensure that these practices are meaningful and impactful.

Finally, the study underscores the value of reflective practices in learning. Retrospectives and feedback sessions should be carefully designed to promote constructive and inclusive communication.

6.6 Contribution to Research

This study contributes to the existing literature by providing a phenomenological perspective on communication skill development in Agile learning environments. Synthesising secondary qualitative data, it offers a nuanced understanding of learners' experiences and identifies key mechanisms that facilitate communication development.

7. Conclusion

This study set out to examine how Agile learning environments contribute to the development of communication skills among learners transitioning from individual coding practices to collaborative, team-based workflows. Drawing on a phenomenological analysis of secondary qualitative data, the findings provide strong evidence that Agile methodologies offer a structured and effective framework for fostering communication competencies in software engineering education.

One of the central conclusions of this research is that communication skill development in Agile environments is not incidental but systematically embedded within the practices themselves. Rituals such as daily stand-ups, sprint planning sessions, and retrospectives create regular and purposeful opportunities for students to articulate ideas, engage in dialogue, and reflect on their experiences. These structured interactions help normalise communication as an essential component of the development process, enabling students to gradually build confidence and competence.

Furthermore, the study highlights the significance of iterative and experiential learning processes in shaping communication skills. Agile practices align closely with experiential learning theory, providing continuous cycles of action, reflection, and improvement. Through repeated participation in collaborative tasks, students refine their ability to express technical concepts, provide constructive feedback, and adapt their communication strategies to different contexts. This iterative engagement

not only enhances communication proficiency but also fosters critical thinking and self-awareness.

The role of social interaction and community participation is equally Agile teams function as communities of practice where learning occurs through shared experiences and mutual engagement. As students assume different roles and responsibilities, they develop diverse communication skills and begin to construct professional identities aligned with industry expectations. This transition from peripheral participation to active contribution reflects a deeper integration into collaborative learning cultures.

However, the study also identifies several challenges that must be addressed to fully realise the potential of Agile learning environments. Issues such as unequal participation, communication anxiety, and cultural or linguistic barriers can limit the effectiveness of these environments. Without intentional instructional design and facilitation, some students may remain marginalised, thereby restricting their opportunities for skill development. Addressing these challenges requires inclusive pedagogical strategies, such as structured participation mechanisms, supportive feedback systems, and culturally responsive teaching practices.

From a practical perspective, the findings underscore the need for educators to integrate communication-focused activities into software engineering curricula deliberately. Assessment frameworks should also evolve to recognise communication as a core competency rather than a supplementary

skill. Additionally, instructors play a crucial role in modelling effective communication and creating psychologically safe environments that encourage open dialogue and collaboration.

In conclusion, Agile learning environments represent a powerful pedagogical approach for bridging the gap between technical expertise and essential soft skills. By embedding communication within iterative and collaborative practices, these environments prepare students for the complex interpersonal demands of modern software development. Future research should build on this work by incorporating primary data and exploring contextual variations across different educational and cultural settings.

References

- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., & Thomas, D. (2001). *Manifesto for agile software development*.
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2), 77–101.
- Creswell, J. W., & Poth, C. N. (2018). *Qualitative inquiry and research design: Choosing among five approaches* (4th ed.). Sage.
- Derby, E., & Larsen, D. (2006). *Agile retrospectives: Making good teams great*. Pragmatic Bookshelf.

- Edmondson, A. (1999). Psychological safety and learning behaviour in work teams. *Administrative Science Quarterly*, 44(2), 350–383.
- Hoda, R., Noble, J., & Marshall, S. (2013). Self-organising roles on Agile software development teams. *IEEE Transactions on Software Engineering*, 39(3), 422–444.
- Johnson, D. W., & Johnson, R. T. (2009). An educational psychology success story: Social interdependence theory. *Educational Researcher*, 38(5), 365–379.
- Kitchenham, B., Budgen, D., & Brereton, P. (2017). *Evidence-based software engineering and systematic reviews*. CRC Press.
- Kolb, D. A. (1984). *Experiential learning: Experience as the source of learning and development*. Prentice Hall.
- Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge University Press.
- Lethbridge, T. C., Diaz-Herrera, J., LeBlanc, R. J., & Thompson, J. B. (2007). Improving software practice through education: Challenges and future trends. *IEEE Software*, 24(5), 57–63.
- Mahnič, V. (2012). A capstone course on Agile software development using Scrum. *IEEE Transactions on Education*, 55(1), 99–106.
- Mannan, K.A., & Farhana, K.M. (2026). *The Principles of Qur’anic Research Methodology: Deriving the Process of Knowledge from Revelation*. KMF Publishers. Open Access (CC BY 4.0). DOI: <https://doi.org/10.64907/xkmf.book.pqrm.26.02.12>
- Mercer, N., & Littleton, K. (2007). *Dialogue and the development of children’s thinking: A sociocultural approach*. Routledge.
- Moustakas, C. (1994). *Phenomenological research methods*. Sage.
- Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., & Still, J. (2008). The impact of Agile practices on communication in software development. *Empirical Software Engineering*, 13(3), 303–337.
- Pope-Ruark, R. (2015). Introducing Agile project management strategies in technical and professional communication courses. *Journal of Business and Technical Communication*, 29(1), 112–133.
- Prince, M. (2004). Does active learning work? A review of the research. *Journal of Engineering Education*, 93(3), 223–231.
- Rigby, D. K., Sutherland, J., & Takeuchi, H. (2016). Embracing Agile. *Harvard Business Review*, 94(5), 40–50.
- Robles, M. M. (2012). Executive perceptions of the top 10 soft skills needed in today’s workplace. *Business Communication Quarterly*, 75(4), 453–465.

- Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide*.
- Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson.
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.
- Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity*. Cambridge University Press.
- Williams, L., & Kessler, R. (2002). *Pair programming illuminated*. Addison-Wesley.