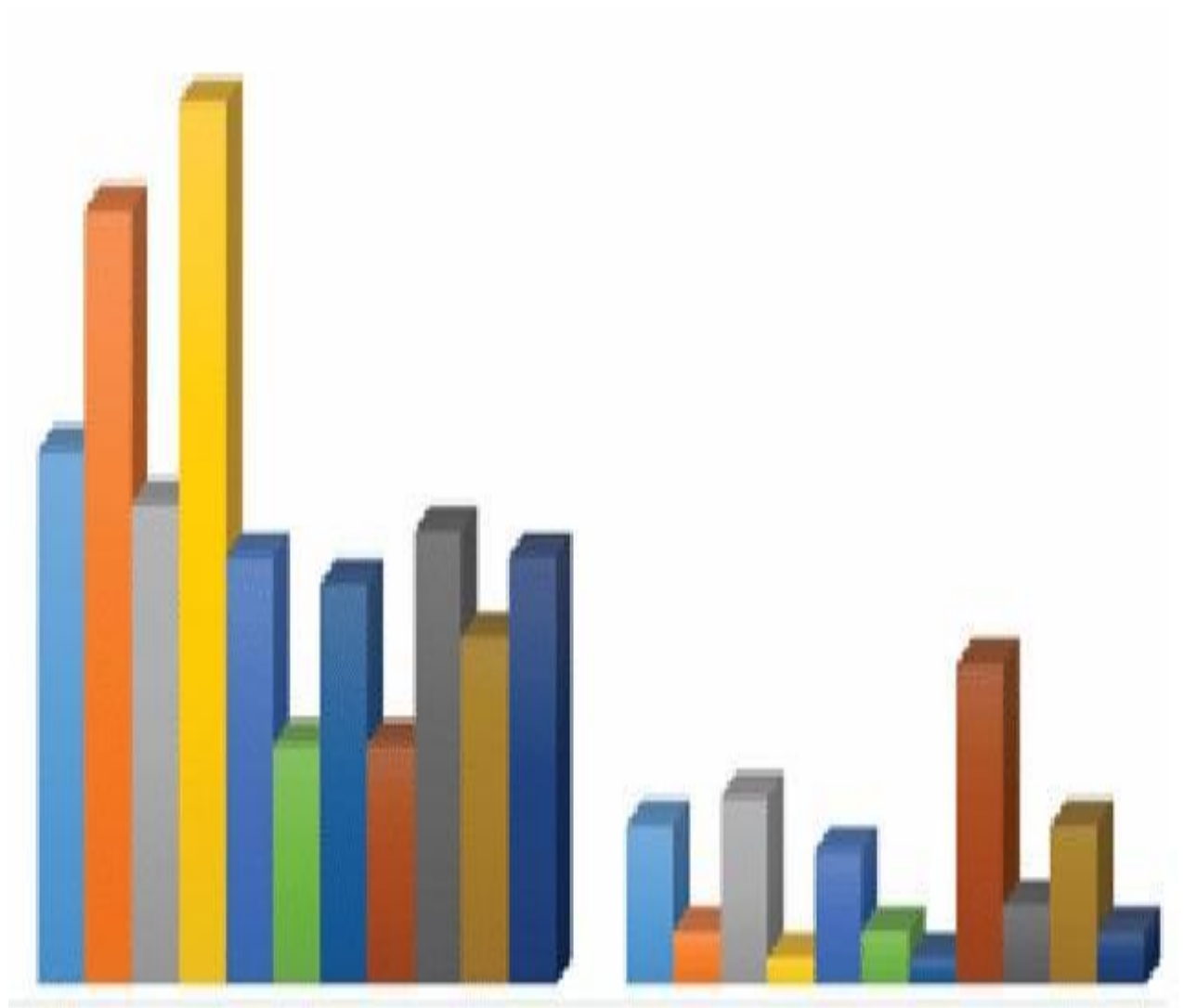


Perception, Motivation and Attitude Studies

Volume: 5 Issue: 2 Year: 2026

ISSN: 2957-4153



Received: 2 May 2026

Revised: 20 June 2026

Accepted: 25 June 2026

DOI: <https://doi.org/10.64907/xkmf.v5i2.pmas.1>

Research Article

Emotional Intelligence in Software Development Teams: A Grounded Theory Study Among CSE Graduates

Mohammad Arman Hossain¹; Kazi Abdul Mannan²

²Department of Computer Science & Information Technology (CSIT), Shanto-Mariam University of Creative Technology Dhaka, Bangladesh.

Correspondence

Email:
armanhossain4550@gmail.com

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

ABSTRACT

Emotional intelligence (EI) has become an increasingly important factor in enhancing the effectiveness of software development teams, particularly in collaborative and high-pressure environments. This study investigates the role of emotional intelligence among Computer Science and Engineering (CSE) graduates through a grounded theory approach based on qualitative secondary data. Drawing on existing literature from behavioural software engineering, organisational psychology, and team dynamics, the research identifies key emotional competencies, self-awareness, self-regulation, empathy, and social skills as critical drivers of team cohesion, communication effectiveness, and project performance. The findings reveal that emotionally intelligent developers are better equipped to manage stress, resolve conflicts, and engage in constructive collaboration, thereby improving overall software quality and productivity. The study further highlights a significant gap in CSE education, where technical skills are emphasised at the expense of emotional and interpersonal development. A grounded theoretical model is proposed, illustrating the cyclical relationship between individual emotional competencies and team-level outcomes. The study concludes by recommending the integration of emotional intelligence training into academic curricula and organisational practices to foster more resilient and effective software development teams.

Keywords: emotional intelligence, software development teams, grounded theory, CSE graduates, team dynamics, behavioural software engineering, soft skills

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 evolution of the global software industry has significantly transformed the nature of work in Computer Science and Engineering (CSE). Traditionally, software development was perceived as a predominantly technical domain, where success depended largely on programming skills, algorithmic competence, and system design expertise. However, the contemporary software development landscape has shifted toward highly collaborative, interdisciplinary, and human-centred environments, where technical proficiency alone is insufficient to ensure project success (Meyer, 2014; Lenberg et al., 2015).

In modern development ecosystems, particularly those adopting Agile and DevOps methodologies, software engineers operate within dynamic teams characterised by continuous communication, iterative problem-solving, and rapid adaptation to changing requirements (Beck et al., 2001). These conditions inherently demand strong interpersonal competencies, including communication, collaboration, and conflict resolution. Within this context, emotional intelligence (EI) has emerged as a critical determinant of team effectiveness and organisational performance.

Emotional intelligence, broadly defined as the ability to perceive, understand, regulate, and utilise emotions in oneself and others, was first conceptualised by Salovey and Mayer (1990) and later popularised by Goleman (1995). Goleman's framework identifies five core components of EI: self-

awareness, self-regulation, motivation, empathy, and social skills. These competencies are increasingly recognised as essential for navigating complex social interactions in professional environments.

The relevance of emotional intelligence in software development is particularly pronounced due to the inherently collaborative nature of the discipline. Software projects often involve cross-functional teams comprising developers, designers, testers, project managers, and stakeholders. These teams must coordinate effectively under conditions of uncertainty, time pressure, and evolving client expectations. Consequently, emotional mismanagement, such as frustration, miscommunication, or interpersonal conflict, can significantly hinder productivity and compromise project outcomes (Lenberg et al., 2015).

Empirical research has demonstrated that emotional states influence cognitive processes relevant to software engineering, including problem-solving, creativity, and decision-making. For instance, Graziotin et al. (2014) found that software developers experiencing positive emotions tend to perform better in analytical tasks compared to those experiencing negative emotional states. Similarly, studies in behavioural software engineering suggest that emotional dynamics within teams directly affect code quality, debugging efficiency, and overall project success (Lenberg et al., 2015).

Despite these insights, the integration of emotional intelligence into software engineering education remains limited. CSE curricula typically emphasise technical

subjects such as programming, data structures, and software architecture, often neglecting the development of soft skills and emotional competencies. As a result, many graduates enter the workforce with strong technical abilities but insufficient preparation for the interpersonal demands of team-based development environments (Pinto et al., 2017).

The transition from academic settings to professional software teams presents significant challenges for CSE graduates. In academic contexts, students often work individually or in small groups with limited exposure to real-world collaboration dynamics. In contrast, industry environments require continuous interaction, negotiation, and coordination with diverse stakeholders. This gap between technical training and social competence underscores the need for a more holistic understanding of emotional intelligence in software development contexts.

Moreover, the increasing adoption of remote and distributed work models has further amplified the importance of emotional intelligence. Virtual teams rely heavily on digital communication tools, where non-verbal cues are limited, and misunderstandings can easily arise. In such settings, emotional awareness and empathy become critical for maintaining team cohesion and ensuring effective collaboration (Meyer, 2014).

While the importance of emotional intelligence in organisational settings is well established, there remains a lack of comprehensive theoretical models that explain its specific role within software

development teams, particularly among early-career professionals such as CSE graduates. Existing studies tend to focus on general workplace outcomes or leadership contexts, with limited attention to the unique challenges and dynamics of technical teams.

To address this gap, the present study employs a grounded theory approach to explore the role of emotional intelligence in software development teams. By analysing qualitative secondary data from academic literature and industry reports, this research aims to develop a conceptual framework that explains how emotional intelligence influences team dynamics, collaboration, and performance.

Specifically, the study seeks to answer the following research questions:

- How does emotional intelligence manifest in software development teams composed of CSE graduates?
- What are the key emotional competencies that influence team effectiveness?
- How can a grounded theory model explain the relationship between emotional intelligence and software development outcomes?

By addressing these questions, this study contributes to both academic and practical domains. Academically, it advances the understanding of emotional intelligence within behavioural software engineering. Practically, it provides insights for educators and organisations seeking to enhance team performance through the development of emotional competencies.

2. Literature Review

Emotional intelligence (EI) has been extensively studied across psychology, organisational behaviour, and management sciences. The foundational work of Salovey and Mayer (1990) conceptualised EI as a subset of social intelligence involving the ability to monitor one's own and others' emotions, discriminate among them, and use this information to guide thinking and behaviour. This ability-based model emphasises cognitive processing of emotional information.

Goleman (1995) expanded this conceptualisation by introducing a mixed model that integrates emotional competencies with personality traits and behavioural skills. His framework identifies five key dimensions: self-awareness, self-regulation, motivation, empathy, and social skills. These components have been widely adopted in organisational research due to their practical relevance.

Subsequent studies have demonstrated that emotional intelligence is positively associated with job performance, leadership effectiveness, and team collaboration (Boyatzis, 2018). Individuals with high EI are better equipped to manage stress, navigate social complexities, and make informed decisions, making EI a critical factor in workplace success.

2.1 Emotional Intelligence in Organisational and Team Contexts

In organisational settings, emotional intelligence plays a pivotal role in shaping team dynamics and interpersonal relationships. Teams characterised by high

emotional intelligence tend to exhibit greater cohesion, trust, and psychological safety (Edmondson, 1999). Psychological safety, defined as a shared belief that the team is safe for interpersonal risk-taking, is essential for fostering open communication and innovation.

Research indicates that emotionally intelligent team members are more effective in conflict resolution, as they can recognise emotional triggers and respond constructively (Jordan & Troth, 2004). This ability reduces the likelihood of destructive conflicts and promotes collaborative problem-solving.

Furthermore, EI contributes to effective leadership within teams. Leaders with high emotional intelligence can inspire, motivate, and guide team members, enhancing overall performance (Goleman, 1998). In software development contexts, where leadership is often distributed and informal, these competencies are particularly valuable.

2.2 Behavioural Software Engineering and Emotional Factors

Behavioural software engineering (BSE) is an emerging field that examines the human and social aspects of software development (Lenberg et al., 2015). BSE research highlights the importance of psychological and emotional factors in influencing developer behaviour and project outcomes.

Lenberg et al. (2015) identified several behavioural factors affecting software development, including motivation, stress, communication, and teamwork. These factors are closely linked to emotional intelligence, suggesting that EI is a

foundational element of effective software engineering practices.

Graziotin et al. (2014) provided empirical evidence that developers' emotional states significantly impact their problem-solving abilities. Their study found that happier developers perform better in analytical tasks, indicating a direct relationship between emotional well-being and cognitive performance.

Similarly, Muller and Fritz (2015) explored the emotional experiences of software developers and found that emotions fluctuate frequently during development activities, influenced by factors such as debugging challenges, team interactions, and time pressure. These findings underscore the dynamic nature of emotions in software engineering contexts.

2.3 Emotional Intelligence in Software Development Teams

Software development teams operate in complex environments that require continuous interaction and coordination. Agile methodologies, such as Scrum and Extreme Programming, emphasise collaboration, customer involvement, and adaptability (Beck et al., 2001). These methodologies inherently rely on effective communication and emotional competence.

Studies have shown that teams with higher levels of emotional intelligence demonstrate improved collaboration and productivity (Pinto et al., 2017). EI facilitates clear communication, reduces misunderstandings, and enhances mutual understanding among team members.

Empathy, a core component of EI, is particularly important in software development. It enables developers to understand user needs, collaborate effectively with colleagues, and respond to feedback constructively. Empathetic developers are more likely to create user-centred solutions and maintain positive team relationships.

Conflict is an inevitable aspect of team-based work, especially in high-pressure environments such as software development. Emotional intelligence plays a crucial role in managing conflicts by enabling individuals to regulate their emotions and engage in constructive dialogue (Jordan & Troth, 2004).

2.4 Challenges Faced by CSE Graduates

CSE graduates often face difficulties adapting to professional software development environments due to a lack of exposure to real-world team dynamics. Academic programs typically focus on technical skills, with limited emphasis on interpersonal and emotional competencies (Pinto et al., 2017).

This gap can lead to challenges such as:

- Ineffective communication
- Difficulty handling feedback
- Poor conflict management
- Reduced collaboration

Moreover, the transition to industry involves working with diverse teams, including individuals from different cultural and professional backgrounds. Emotional intelligence is essential for navigating these

complexities and building effective working relationships.

2.5 Remote Work and Emotional Intelligence

The rise of remote and distributed software development teams has introduced new challenges related to communication and collaboration. Virtual environments limit non-verbal cues, making it more difficult to interpret emotions and intentions (Meyer, 2014).

In such contexts, emotional intelligence becomes even more critical. Team members must rely on empathy, active listening, and clear communication to maintain cohesion and avoid misunderstandings. Research suggests that emotionally intelligent individuals are better equipped to adapt to remote work environments and maintain productivity.

2.6 Gaps in the Literature

Despite the growing body of research on emotional intelligence and software development, several gaps remain:

- Limited focus on early-career professionals, particularly CSE graduates
- Lack of grounded theory-based models explaining EI in software teams
- Insufficient integration of EI into software engineering education

This study addresses these gaps by synthesising existing literature and developing a grounded theoretical framework.

3. Theoretical Framework

The present study is grounded in an integrative theoretical framework that combines Emotional Intelligence Theory, Social Constructivism, and Grounded Theory Methodology to examine the role of emotional intelligence (EI) in software development teams among Computer Science and Engineering (CSE) graduates. This multi-theoretical approach allows for a comprehensive understanding of both the individual and collective dimensions of emotional processes in collaborative technical environments.

3.1 Emotional Intelligence Theory

Emotional intelligence theory serves as the primary conceptual foundation for this study. Initially proposed by Salovey and Mayer (1990), EI is defined as the ability to perceive, assess, and manage emotions in oneself and others. This ability-based model emphasises cognitive processing of emotional information and highlights the role of emotions in guiding rational decision-making.

Building upon this foundation, Goleman (1995, 1998) introduced a mixed model of emotional intelligence that incorporates both cognitive abilities and behavioural competencies. According to Goleman, EI comprises five key dimensions:

- **Self-awareness** – the ability to recognise and understand one's emotions
- **Self-regulation** – the capacity to manage emotional responses effectively

- **Motivation** – the intrinsic drive to achieve goals
- **Empathy** – the ability to understand others' emotions
- **Social skills** – the ability to manage relationships and build networks

These components are particularly relevant in software development contexts, where individuals must collaborate, communicate, and resolve conflicts in team-based settings. Empirical research suggests that individuals with high EI demonstrate superior teamwork, adaptability, and leadership capabilities (Boyatzis, 2018).

In the context of this study, emotional intelligence is conceptualised as a multidimensional construct that influences both individual behaviour and team dynamics. Each EI component contributes to different aspects of team functioning. For example, self-regulation helps developers manage stress during tight deadlines, while empathy facilitates understanding among team members, reducing interpersonal conflicts.

3.2 Social Constructivism

The second theoretical pillar of this study is social constructivism, which emphasises the role of social interaction in the construction of knowledge and meaning. Rooted in the work of Vygotsky (1978), social constructivism posits that learning and development occur through collaborative processes and shared experiences.

In software development teams, knowledge is not created in isolation but emerges through continuous interaction among team members. Activities such as code reviews,

pair programming, sprint meetings, and collaborative debugging exemplify this process. Emotional intelligence plays a crucial role in facilitating these interactions by enabling individuals to communicate effectively, interpret social cues, and respond appropriately to others' perspectives.

From a social constructivist perspective, emotional intelligence can be understood as a mediating factor that enhances collaborative knowledge construction. For instance, empathy allows developers to appreciate diverse viewpoints, while social skills enable constructive dialogue. These competencies contribute to the development of shared understanding and collective problem-solving.

Furthermore, psychological safety, an important concept within team dynamics, is closely aligned with social constructivist principles. Edmondson (1999) defines psychological safety as a shared belief that the team is safe for interpersonal risk-taking. Emotional intelligence contributes to the creation of such environments by fostering trust, openness, and mutual respect.

Thus, social constructivism provides a theoretical lens for understanding how emotional intelligence influences team-level processes, including communication, collaboration, and knowledge sharing.

3.3 Behavioural Software Engineering Perspective

To contextualise emotional intelligence within the software development domain, this study also draws on the principles of Behavioural Software Engineering (BSE). BSE focuses on the cognitive, behavioural,

and social aspects of software engineering processes (Lenberg et al., 2015).

BSE research highlights that software development is not purely a technical activity but a socio-technical process influenced by human behaviour, emotions, and interactions. Factors such as motivation, stress, and communication significantly affect developer performance and project outcomes (Graziotin et al., 2014).

Integrating BSE into the theoretical framework allows this study to link emotional intelligence with specific software engineering practices. For example:

- Emotional awareness influences debugging efficiency
- Empathy enhances user-centred design
- Social skills improve team coordination

This integration reinforces the argument that emotional intelligence is not peripheral but central to effective software development.

3.4 Grounded Theory Perspective

The final component of the theoretical framework is grounded theory, which provides the methodological foundation for theory development. Grounded theory, introduced by Glaser and Strauss (1967), emphasises the generation of theory from data rather than the testing of pre-existing hypotheses.

In this study, grounded theory is used to synthesise qualitative secondary data and develop a conceptual model explaining the role of emotional intelligence in software development teams. The approach involves

iterative coding and constant comparison to identify patterns, relationships, and core categories.

The grounded theory perspective aligns well with the exploratory nature of this research. Given the limited theoretical models specifically addressing EI in software teams, an inductive approach is necessary to uncover underlying mechanisms and relationships.

3.5 Integrated Conceptual Model

By integrating emotional intelligence theory, social constructivism, and behavioural software engineering, this study proposes a conceptual framework in which:

- **Individual-level EI competencies** influence
- **Interpersonal interactions and communication**, which shape
- **Team dynamics and collaborative processes**, ultimately affecting
- **Software development outcomes**

This framework highlights the multilevel impact of emotional intelligence, bridging the gap between individual behaviour and team performance.

4. Methodology

This study adopts a qualitative research design grounded in the principles of grounded theory. Qualitative research is particularly suitable for exploring complex social phenomena, such as emotional intelligence and team dynamics, where contextual understanding and interpretive analysis are essential (Creswell, 2014).

Grounded theory was selected as the primary methodological approach because it enables the development of theory directly from data. Unlike deductive approaches, which test predefined hypotheses, grounded theory allows for the emergence of new insights and conceptual frameworks (Glaser & Strauss, 1967).

4.1 Data Source and Secondary Data Approach

The study relies on secondary qualitative data, which includes previously published research and documented empirical findings. Secondary data analysis is an established method in qualitative research, particularly when primary data collection is impractical or when the objective is to synthesise existing knowledge (Johnston, 2017).

Data Sources: Data were collected from reputable academic and professional databases, including:

- IEEE Xplore
- ACM Digital Library
- Scopus
- Google Scholar

The selected sources include:

- Peer-reviewed journal articles
- Conference proceedings
- Systematic literature reviews
- Industry reports

Inclusion Criteria: The following criteria were used to select relevant studies:

- Focus on emotional intelligence, behavioural factors, or team dynamics

- Relevance to software development or technical teams
- Empirical or theoretical contribution
- Publication within the last 15 years (with some foundational exceptions)

Exclusion Criteria

Studies were excluded if they:

- Focused solely on technical aspects without human factors
- Lacked methodological rigour
- Were not available in full text

4.2 Data Analysis Procedures

The analysis followed the three main stages of grounded theory coding:

Open Coding: In the first stage, data were examined line-by-line to identify key concepts and themes. Codes such as “emotional awareness,” “team conflict,” “communication barriers,” and “developer motivation” were generated.

Axial Coding: During axial coding, relationships between categories were established. For example:

- Emotional awareness was linked to self-regulation
- Empathy was associated with conflict resolution
- Communication skills were connected to team cohesion

This stage involved grouping related codes into broader categories and identifying causal relationships.

Selective Coding: In the final stage, a core category was identified, and a theoretical model was developed. The core category,

emotional intelligence as a catalyst for collaborative efficiency, integrates all other categories and explains the central phenomenon.

The constant comparative method was used throughout the analysis to refine categories and ensure consistency (Glaser & Strauss, 1967).

4.3 Ensuring Research Rigour

To enhance the credibility and trustworthiness of the study, several strategies were employed:

Triangulation: Data were collected from multiple sources to ensure a comprehensive understanding of the phenomenon. This reduces bias and increases validity.

Theoretical Saturation: Data analysis continued until no new themes or categories emerged, indicating theoretical saturation.

Transparency: The coding process and analytical steps were documented to ensure transparency and replicability.

Peer-Reviewed Sources: Only high-quality, peer-reviewed studies were included to ensure reliability.

4.4 Ethical Considerations

As this 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 plagiarism
- Ensuring accurate representation of original findings (Mannan & Farhana, 2026)

4.5 Limitations of the Methodology

While the use of secondary data provides valuable insights, it also presents certain limitations:

- Lack of control over data quality and context
- Potential bias in original studies
- Limited ability to capture real-time experiences of CSE graduates

Despite these limitations, the methodology is appropriate for developing a theoretical framework and identifying patterns across multiple studies.

4.6 Justification of Methodological Choice

The combination of qualitative research, grounded theory, and secondary data analysis is particularly suitable for this study because:

- It allows for theory generation in an underexplored area
- It synthesises diverse perspectives and findings
- It provides a holistic understanding of emotional intelligence in software teams

5. Findings and Analysis

The grounded theory analysis of secondary qualitative data revealed a complex and interrelated set of categories that collectively explain how emotional intelligence (EI) influences software development teams composed of Computer Science and Engineering (CSE) graduates. Through systematic coding procedures, open, axial, and selective coding, this study identified

five major categories: emotional awareness, self-regulation, empathy, communication effectiveness, and team cohesion. These categories are interconnected and collectively contribute to a central phenomenon: the enhancement of collaborative efficiency and project performance in software development environments.

5.1 Emotional Awareness as the Foundational Competency

Emotional awareness emerged as the most fundamental component of emotional intelligence within software development teams. It refers to the ability of individuals to recognise and understand their own emotional states and how these states influence their behaviour and interactions (Goleman, 1995).

In software engineering contexts, emotional awareness is particularly important due to the cognitively demanding and often stressful nature of development tasks. Developers frequently encounter challenges such as debugging complex code, meeting tight deadlines, and adapting to changing requirements. These situations can evoke a range of emotional responses, including frustration, anxiety, and satisfaction.

The analysis indicates that developers with high emotional awareness are better equipped to manage these emotional fluctuations. For instance, recognising feelings of frustration during debugging allows individuals to take proactive steps, such as seeking assistance or taking breaks, thereby preventing emotional escalation and cognitive overload. This finding aligns with Graziotin et al. (2014),

who demonstrated that emotional states significantly influence developers' problem-solving abilities.

Moreover, emotional awareness contributes to improved decision-making. By understanding their emotional biases, developers can make more rational and objective decisions, reducing the likelihood of errors and enhancing code quality. This supports the argument that emotions are not merely peripheral but integral to cognitive processes in software engineering (Lenberg et al., 2015).

5.2 Self-Regulation and Stress Management

Closely linked to emotional awareness is self-regulation, which involves the ability to control and manage emotional responses. In the context of software development teams, self-regulation plays a critical role in maintaining professionalism and ensuring effective collaboration.

The data reveal that self-regulation is particularly important in high-pressure situations, such as approaching project deadlines or dealing with critical bugs. Developers who can regulate their emotions are less likely to engage in impulsive behaviours, such as blaming colleagues or making hasty decisions. Instead, they adopt constructive coping strategies, such as problem-focused thinking and collaborative troubleshooting.

This finding is consistent with the broader literature on emotional intelligence, which highlights self-regulation as a key determinant of workplace performance (Boyatzis, 2018). In software teams, self-

regulation contributes to a stable and supportive work environment, reducing the negative impact of stress and conflict.

Furthermore, self-regulation enhances resilience. Developers who can manage stress effectively are more likely to persist in the face of challenges and maintain productivity over time. This is particularly relevant in Agile environments, where continuous iteration and rapid feedback loops require sustained engagement and adaptability (Beck et al., 2001).

5.3 Empathy and Interpersonal Understanding

Empathy emerged as a central category influencing team dynamics and collaboration. It refers to the ability to understand and share the feelings of others, enabling individuals to respond appropriately to social and emotional cues (Goleman, 1995).

In software development teams, empathy facilitates effective communication and reduces interpersonal conflicts. Developers often work with colleagues from diverse backgrounds, including different technical specialisations, cultures, and levels of experience. Empathy allows team members to appreciate these differences and adapt their communication styles accordingly.

The analysis indicates that empathetic developers are more likely to engage in active listening, provide constructive feedback, and support their colleagues. This fosters a positive team climate characterised by trust and mutual respect. Edmondson (1999) emphasises that such environments are

essential for psychological safety, which in turn promotes learning and innovation.

Empathy also plays a critical role in user-centred design. Developers who can understand the needs and perspectives of end-users are better equipped to create software that meets user expectations. This highlights the broader impact of emotional intelligence beyond team interactions, extending to product quality and customer satisfaction.

5.4 Communication Effectiveness as a Mediating Factor

Communication effectiveness emerged as a key mediating factor linking individual emotional competencies to team outcomes. Effective communication involves not only the exchange of information but also the ability to convey emotions, intentions, and feedback clearly and constructively.

The findings suggest that emotionally intelligent developers are more effective communicators. They can articulate their ideas clearly, listen actively, and respond appropriately to feedback. This reduces misunderstandings and enhances coordination among team members.

In Agile environments, where frequent communication is essential, the importance of communication effectiveness is further amplified. Practices such as daily stand-up meetings, sprint reviews, and retrospectives rely heavily on open and transparent communication (Beck et al., 2001).

Moreover, communication effectiveness is closely linked to conflict resolution. Teams inevitably encounter disagreements,

particularly in decision-making processes. Emotionally intelligent individuals can navigate these conflicts constructively by expressing their perspectives respectfully and considering alternative viewpoints. This aligns with Jordan and Troth (2004), who found that emotional intelligence improves team problem-solving by facilitating constructive conflict management.

5.5 Team Cohesion and Psychological Safety

Team cohesion, defined as the degree of unity and collaboration among team members, emerged as a critical outcome of emotional intelligence. High levels of EI contribute to stronger interpersonal relationships, trust, and a sense of belonging within the team.

The analysis indicates that emotionally intelligent teams are more likely to develop psychological safety, which encourages members to share ideas, take risks, and admit mistakes without fear of negative consequences (Edmondson, 1999). This is particularly important in software development, where innovation and continuous improvement are essential.

Team cohesion also enhances collective problem-solving. When team members trust each other and communicate effectively, they can leverage diverse perspectives and expertise to address complex challenges. This leads to higher-quality solutions and improved project outcomes.

5.6 Axial Relationships and Process Dynamics

The axial coding process revealed several key relationships among the identified categories:

- Emotional awareness → Self-regulation → Reduced stress and improved decision-making
- Empathy → Effective communication → Conflict resolution
- Social skills → Team cohesion → Enhanced collaboration

These relationships suggest that emotional intelligence operates as a **dynamic process** rather than a static trait. Individual competencies interact with team-level processes to produce outcomes that reinforce the development of EI over time.

5.7 Core Category and Grounded Theory Model

The selective coding process identified the core category:

“Emotional Intelligence as a Catalyst for Collaborative Efficiency in Software Development Teams.”

This core category integrates all other findings and forms the basis of the grounded theory model. The model proposes a cyclical process:

- Development of individual EI competencies
- Improvement in interpersonal interactions
- Enhancement of team dynamics

- Achievement of better project outcomes
- Reinforcement of EI through experience

This cyclical model highlights the self-reinforcing nature of emotional intelligence in software development environments.

6. Discussion

The findings of this study provide significant insights into the role of emotional intelligence in software development teams, particularly among CSE graduates. By integrating the results with existing theoretical and empirical literature, this section offers a deeper interpretation of the implications for education, industry, and future research.

6.1 Reframing Software Engineering as a Socio-Technical Discipline

One of the key implications of this study is the need to reconceptualise software engineering as a socio-technical discipline. Traditional approaches have emphasised technical skills, often overlooking the human and emotional dimensions of development work. However, the findings clearly demonstrate that emotional intelligence is integral to effective team functioning and project success.

This perspective aligns with the principles of Behavioural Software Engineering, which emphasise the importance of human factors in software development (Lenberg et al., 2015). Emotional intelligence serves as a bridge between technical expertise and social

interaction, enabling developers to navigate the complexities of collaborative work environments.

6.2 Implications for CSE Education

The findings highlight a significant gap in CSE education: the lack of emphasis on emotional and interpersonal skills. While technical competencies remain essential, they are insufficient for preparing graduates for real-world software development environments.

Integrating emotional intelligence into CSE curricula could take several forms:

- Courses on communication and teamwork
- Training in conflict resolution and emotional regulation
- Collaborative projects that simulate real-world scenarios

Such interventions would help students develop the emotional competencies necessary for effective teamwork. This is consistent with Pinto et al. (2017), who argue that non-technical skills are critical for developer success.

6.3 Organisational Implications

From an organisational perspective, the findings suggest that companies should prioritise emotional intelligence in both recruitment and professional development. Traditional hiring practices often focus on technical skills, but incorporating EI assessments could provide a more holistic evaluation of candidates.

Organisations can also implement training programs to enhance emotional intelligence

among employees. These programs may include:

- Workshops on communication and empathy
- Coaching and mentoring initiatives
- Team-building activities

Additionally, fostering a culture of psychological safety is essential for maximising the benefits of emotional intelligence. Leaders play a crucial role in creating such environments by modelling emotionally intelligent behaviour and encouraging open communication (Edmondson, 1999).

6.4 Emotional Intelligence in Agile and Remote Work Contexts

The increasing adoption of Agile methodologies and remote work models has further amplified the importance of emotional intelligence. Agile practices rely heavily on collaboration, adaptability, and continuous feedback, all of which require strong emotional competencies (Beck et al., 2001).

In remote work environments, the absence of face-to-face interaction can lead to communication challenges and misunderstandings. Emotional intelligence helps mitigate these challenges by enabling individuals to interpret digital cues, express themselves clearly, and maintain positive relationships.

Meyer (2014) emphasises that cultural and communication differences can be particularly pronounced in global teams. Emotional intelligence facilitates cross-

cultural understanding and enhances collaboration in such contexts.

6.5 Theoretical Contributions

This study contributes to the existing literature by developing a grounded theory model that explains the role of emotional intelligence in software development teams. Unlike previous studies that focus on isolated aspects of EI, this research provides a comprehensive framework that integrates individual competencies, team dynamics, and project outcomes.

The model also extends emotional intelligence theory by demonstrating its applicability in technical domains. It highlights the multidimensional and dynamic nature of EI, emphasising its role as both an individual attribute and a collective process.

6.6 Limitations and Future Research Directions

While this study provides valuable insights, it is not without limitations. The reliance on secondary data may limit the depth of contextual understanding, and the findings may not fully capture the experiences of CSE graduates in specific organisational settings.

Future research could address these limitations by:

- Conducting primary qualitative studies (e.g., interviews, focus groups)
- Using quantitative methods to validate the proposed model
- Exploring cultural differences in emotional intelligence

Additionally, longitudinal studies could examine how emotional intelligence develops over time and its long-term impact on career progression.

6.7 Practical Recommendations

Based on the findings, several practical recommendations can be made:

- **Educational Reform:** Integrate EI training into CSE curricula
- **Organisational Training:** Develop EI-focused professional development programs
- **Leadership Development:** Encourage emotionally intelligent leadership
- **Team Practices:** Promote open communication and psychological safety

These recommendations aim to enhance both individual and organisational performance.

7. Conclusion

This study has explored the critical role of emotional intelligence (EI) in software development teams, with a particular focus on Computer Science and Engineering (CSE) graduates transitioning into professional environments. By employing a grounded theory approach based on qualitative secondary data, the research provides a comprehensive understanding of how emotional competencies influence individual behaviour, team dynamics, and overall project performance.

The findings demonstrate that emotional intelligence is not merely a supplementary skill but a foundational component of

effective software engineering practice. Core EI dimensions, such as self-awareness, self-regulation, empathy, and social skills, were found to significantly enhance communication, conflict resolution, and collaboration within teams. These competencies enable developers to navigate the complex interpersonal dynamics inherent in modern software development environments, particularly those characterised by Agile methodologies and distributed work structures.

A key contribution of this study is the development of a grounded theoretical model that conceptualises emotional intelligence as a catalyst for collaborative efficiency. The model illustrates a cyclical process in which individual emotional competencies foster improved team interactions, leading to enhanced project outcomes, which in turn reinforce the development of emotional intelligence over time. This dynamic perspective highlights the importance of continuous learning and adaptation in both individual and organisational contexts.

The study also identifies a critical gap in CSE education, where the emphasis on technical skills often overshadows the development of emotional and interpersonal competencies. Addressing this gap requires a paradigm shift in educational and professional training frameworks. Integrating emotional intelligence into curricula, promoting experiential learning, and fostering collaborative environments can better prepare graduates for the realities of the software industry.

From an organisational perspective, the findings underscore the need for companies

to prioritise emotional intelligence in recruitment, training, and leadership development. Creating psychologically safe and emotionally supportive work environments can significantly enhance team performance and innovation.

In conclusion, emotional intelligence is a vital determinant of success in software development teams. Future research should build upon this study by incorporating primary data and exploring cross-cultural and longitudinal dimensions of emotional intelligence in software engineering contexts.

References

- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., & Thomas, D. (2001). *Manifesto for Agile Software Development*. <https://agilemanifesto.org>
- Boyatzis, R. E. (2018). The behavioural level of emotional intelligence and its measurement. *Journal of Organisational Behaviour*, 39(2), 1–15.
- Creswell, J. W. (2014). *Research design: Qualitative, quantitative, and mixed methods approaches* (4th ed.). Sage Publications.
- Edmondson, A. (1999). Psychological safety and learning behaviour in work teams. *Administrative Science Quarterly*, 44(2), 350–383.
- Glaser, B. G., & Strauss, A. L. (1967). *The discovery of grounded theory: Strategies for qualitative research*. Aldine Publishing.
- Goleman, D. (1995). *Emotional intelligence: Why it can matter more than IQ*. Bantam Books.
- Goleman, D. (1998). *Working with emotional intelligence*. Bantam Books.
- Graziotin, D., Wang, X., & Abrahamsson, P. (2014). Happy software developers solve problems better: The link between affect and problem-solving performance. *PeerJ*, 2, e289. <https://doi.org/10.7717/peerj.289>
- Jordan, P. J., & Troth, A. C. (2004). Managing emotions during team problem solving: Emotional intelligence and conflict resolution. *Journal of Organisational Behaviour*, 25(2), 195–218.
- Johnston, M. P. (2017). Secondary data analysis: A method whose time has come. *Qualitative and Quantitative Methods in Libraries*, 3(3), 619–626.
- Lenberg, P., Feldt, R., & Wallgren, L. G. (2015). Behavioural software engineering: A definition and systematic literature review. *Journal of Systems and Software*, 107, 15–37.
- 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>

- Meyer, B. (2014). *Agile! The good, the hype and the ugly*. Springer.
- Muller, S. C., & Fritz, T. (2015). Stuck and frustrated or in flow and happy: Sensing developers' emotions and progress. In *Proceedings of the 37th International Conference on Software Engineering* (pp. 688–699). IEEE.
- Pinto, G., Wiese, I., & Castor, F. (2017). What makes a good developer? An empirical study. *IEEE Software*, 34(1), 44–49.
- Salovey, P., & Mayer, J. D. (1990). Emotional intelligence. *Imagination, Cognition and Personality*, 9(3), 185–211.
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.