

# Investigating Students' Teamwork Experiences in Collaborative Software Projects

## Computer Science Students' Perceived Conflicts and Resolutions

**Axl Jacob Mostert**[1]
**Supervisors: Xiaoqi Feng**[1]**, Rafael Bidarra**[1]**, Martin Skrodzki**[1]
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 21, 2026

Name of the student: Axl Jacob Mostert
Final project course: CSE3000 Research Project
Thesis committee: Xiaoqi Feng, Rafael Bidarra, Martin Skrodzki, Jan van Gemert

An electronic version of this thesis is available at https://repository.tudelft.nl/.

**Abstract**

This qualitative study explores how final-year Computer Science students perceive and manage conflicts during collaborative software projects. While these projects simulate professional teamwork, they frequently introduce interpersonal and organizational challenges. Using semi-structured interviews and Reflexive Thematic Analysis, this research investigates the subjective experiences and resolution strategies of students navigating team conflicts. The findings reveal three primary themes regarding conflict perception and resolution outcomes. First, teams generally manage minor task-oriented disagreements constructively through clear communication, resulting in incidental friction with no lasting impact. Second, prolonged behavioral conflicts, such as teammate absenteeism, can be resolved positively when managed constructively; these experiences often lead to personal growth, leadership, and communication skills. Third, escalating relationship conflicts, characterized by disruptive behavior and ineffective escalation processes, lead to significant negative consequences, including academic delays and emotional strain. The study concludes that while students demonstrate resilience and leverage routine teamwork obstacles to develop essential professional competencies, they struggle to navigate severe interpersonal conflicts without adequate external mediation. To optimize collaborative learning, software engineering curricula could integrate dedicated conflict management workshops, and universities must ensure accessible, transparent institutional support structures for complex interpersonal dynamics

# 1 Introduction

Modern software engineering projects require collaboration between people with diverse technical and interpersonal expertise. As software systems become increasingly complex, communication and teamwork skills have become essential rather than supplementary (Marques & Ochoa, 2014). In response to these industry demands, software engineering education increasingly incorporates collaborative, project-based learning experiences intended to simulate professional teamwork settings (Marques & Ochoa, 2014).

At TU Delft, the BSc Computer Science and Engineering (CSE) curriculum provides students with teamwork experience through initiatives such as the Object-Oriented Programming Project (OOPP), the Software Project (SP), and various smaller collaborative assignments. While these collaborative software projects are associated with educational benefits, such as improved problem-solving and peer learning (Marques & Ochoa, 2014), student software teams regularly encounter interpersonal and organizational challenges.

Existing research shows that communication skills significantly influence engineering employability (Wu et al., 2023), yet students often struggle with the practical challenges of interpersonal communication and conflict resolution (Dorić et al., 2025; Iacob & Faily, 2019). Despite the recognized importance of constructive teamwork, existing literature primarily focuses on teamwork and conflict through quantitative performance metrics or final academic grades (Vanhanen et al., 2017). Consequently, there remains a deficiency in understanding the qualitative experiences of students when navigating conflicts within teamwork environments.

Understanding these nuances is critical for educators to optimize collaborative learning environments and better equip students for the software industry. Therefore, the purpose of this qualitative study is to explore how final-year BSc students experience and manage conflicts during collaborative assignments. Specifically, this study investigates the following research question:

> ***RQ1**: How do final-year BSc Computer Science students perceive teamwork conflicts and resolve these conflicts in collaborative software projects?*

By conducting semi-structured interviews and thematic analysis, this study aims to define themes that emerge when students come into contact with conflict and provide insights that can be used to optimize collaborative learning environments and better prepare students for the software industry. Equipped with these insights, educators can move beyond generic teamwork advice to develop systems that actively reward constructive conflict management when such conditions occur.

# 2 Literature Review

This section defines teamwork and its application in software engineering education, while examining the nature of conflict, recurring challenges in student teams, and frameworks for analyzing these dynamics. Finally, it identifies the research gaps this study addresses.

## 2.1 Definition and Contextualization of Teamwork in Software Engineering Education

Researchers across organizational psychology, management, and engineering education have proposed various definitions of teamwork that emphasize coordination, communication, and shared responsibility (Cohen & Bailey, 1997; Katzenbach & Smith, 1993). Recent perspectives conceptualize teamwork as an adaptive and evolving interaction process rather than a static organizational structure. Kozlowski and Ilgen (2006) argue that effective teamwork requires continuous coordination of knowledge, skills, and interpersonal behaviors in response to changing project demands. These perspectives highlight the importance of interpersonal dynamics and adaptability, particularly within collaborative environments characterized by uncertainty and changing requirements. A more recent study in engineering education characterizes teamwork as the capacity for individuals to coordinate their efforts toward a shared project in a constructive, healthy, and effective manner (Kimpton & Maynard, 2025).

Within software engineering education, teamwork is often used in project-based learning environments to simulate professional software development practices. Marques and Ochoa (2014) argue that collaborative software projects provide students with opportunities to develop technical, organizational, and communication skills simultaneously. Similarly, Dringenberg and Purzer (2018) emphasize that engineering teamwork involves processes such as collaboration, iteration, reasoning, and growth throughout project development. In this context, teamwork involves not only technical cooperation but also negotiation, accountability, and interpersonal coordination between team members.

These collaborative learning environments are widely used in software engineering education because they help students develop both technical and professional skills. Project-based collaboration has been associated with improved problem-solving skills, peer learning, and preparation for the software engineering industry (Marques & Ochoa, 2014). Furthermore, teamwork experiences allow students to develop communication and coordination skills that are increasingly valued within professional engineering environments (Wu et al., 2023). Because modern software engineering projects rarely occur in isolation, educational programs increasingly incorporate group assignments intended to simulate workplace collaboration.

Despite the educational benefits associated with collaborative software projects, teamwork experiences within student environments are not always positive or effective. Collaborative projects frequently introduce organizational difficulties, communication breakdowns, and interpersonal tensions that may complicate both learning experiences and project outcomes (Iacob & Faily, 2019; Vanhanen et al., 2017). Consequently, understanding the challenges associated with student teamwork has become an increasingly important area of software engineering education research.

## 2.2 Definition and Contextualization of Conflict in Software Engineering Education

Conflict is commonly understood as a situation in which individuals or groups perceive incompatibilities between goals, interests, behaviors, or expectations. Deutsch (1973) defines conflict as the presence of incompatible activities between parties, while Thomas (1992) emphasizes that conflict emerges when one party perceives another as negatively affecting something they value. These definitions highlight that conflict is not purely objective but is strongly dependent on individual interpretation and perception.

Importantly, conflict is not necessarily destructive. Deutsch (1973) argues that the consequences of conflict depend heavily on whether individuals adopt cooperative or competitive approaches toward resolution. Certain forms of disagreement may encourage critical discussion, reflection, and improved decision-making, while other forms may escalate interpersonal tension and reduce collaboration effectiveness. Consequently, contemporary teamwork research often distinguishes between productive and destructive forms of conflict rather than treating all conflict as inherently negative (Jehn, 1995).

Within collaborative software engineering projects, conflict is generally considered unavoidable due to differences in technical perspectives, communication styles, work habits, and expectations regarding contribution (Kozlowski & Ilgen, 2006). Because software development requires high levels of coordination and interdependence, even small misunderstandings may influence team dynamics and project organization.

Previous studies have identified several recurring challenges within student software teams. Iacob and Faily (2019) found that students often experience mismatches between expectations and the realities of team-

work, particularly regarding accountability, workload distribution, and communication practices. Similarly, Dorić et al. (2025) reported that teamwork competencies among undergraduate software engineering students are frequently unevenly developed, contributing to coordination difficulties and inconsistent collaboration quality. Vanhanen et al. (2017) further observed that teamwork problems negatively influence both perceived learning outcomes and team satisfaction.

These findings suggest that unmanaged interpersonal conflict within student software projects does more than just lower project grades. It can result in emotional stress, unequal workload distributions, and academic setbacks. Because collaborative projects are intended to simulate real-world software engineering, only grading the final result is insufficient. Understanding exactly how students subjectively experience and navigate conflicts is essential for moving beyond generic teamwork advice and developing actions that support a student's professional and personal communication skills.

### 2.3 Frameworks for Understanding Conflict

One of the most influential frameworks for understanding conflict was created by Jehn (1995), who distinguishes between task, process, and relationship conflicts. Task conflict refers to disagreements regarding ideas, decisions, or work content. Process conflict concerns disagreements about task delegation, coordination, and responsibility distribution. Relationship conflict involves interpersonal tension, emotional friction, or incompatibilities between team members. This framework is valuable because it differentiates between conflicts related to project execution and those related to interpersonal relationships.

Jehn (1995) argues that different forms of conflict may produce different outcomes within teams. Moderate levels of task conflict may stimulate critical thinking and improve decision-making by encouraging members to evaluate alternative perspectives. In contrast, relationship conflict is associated with reduced cohesion, trust, and collaboration effectiveness. Consequently, conflict should not be treated as a singular phenomenon but rather as a multidimensional interaction process with varying consequences.

Additional perspectives further emphasize the social complexity of conflict management within collaborative groups. Deutsch (1973) distinguishes between cooperative and competitive conflict-resolution behaviors, arguing that constructive outcomes depend heavily on how individuals approach disagreement. Within engineering education research, Dringenberg and Purzer (2018) demonstrate that students' teamwork experiences evolve throughout collaborative problem-solving processes, particularly when teams encounter ambiguity or organizational uncertainty. Together, these frameworks provide conceptual tools for interpreting how students perceive and navigate conflict within software engineering projects.

### 2.4 Research Gaps

Existing literature demonstrates that collaborative software engineering education provides professional and educational benefits while introducing interpersonal and organizational challenges. However, much of the existing research emphasizes measurable teamwork outcomes, project performance, or generalized collaboration effectiveness rather than students' subjective experiences and interpretations of conflict situations. While prior studies provide useful categorizations of teamwork behaviors and conflict types, relatively limited research has qualitatively explored how final-year BSc computer science students personally perceive, interpret, and resolve conflicts during collaborative software engineering projects. Addressing this qualitative gap is essential because students who merely survive dysfunctional academic teams without developing resolution strategies often carry these unmanaged communication deficits into the software industry, which can impact their future career performance.

## 3 Methodology

The study uses a qualitative research approach. Since interpersonal conflict involves subjective experiences, communication patterns, and individual interpretations, qualitative methods are appropriate for capturing detailed and context-dependent insights (Creswell, 2013).

## 3.1 Curricular Context

The TU Delft BSc Computer Science and Engineering curriculum contains multiple collaborative software engineering projects throughout the bachelor program. These projects typically require students to coordinate development tasks, communicate regularly within teams, manage deadlines, and solve technical problems.

Because students repeatedly encounter teamwork projects throughout the curriculum, they are likely to develop diverse experiences regarding collaboration and conflict. This curricular structure provides a suitable context for investigating how students perceive and resolve teamwork conflicts. Key collaborative projects relevant to this study include the Object Oriented Programming Project (OOPP) in Year 1 (Q3), which involves team-based software development and coordination; the large-scale Software Project (SP) in Year 2 (Q4), featuring structured teamwork and role division; and some minor programs in Year 3 (Q1 & Q2) that incorporate collaborative software assignments.

## 3.2 Participants

Participants were recruited through contact-sharing within the researchers' own academic network and research group connections. This recruitment strategy was primarily selected due to the limited ten-week time-frame of the study. Therefore, a small size of four students was successfully recruited. Accessing final-year Computer Science students within this time-frame would have been difficult through broader institutional recruitment methods alone. As a result, network-based recruitment provided a practical method for identifying students with the criteria needed for the study.

Final-year students were specifically targeted because they possess experience with collaborative coursework and team-based software projects throughout the curriculum. These students are therefore more likely to have encountered a variety of teamwork conflicts, communication challenges, and conflict-resolution situations compared to earlier-year students.

## 3.3 Data Collection

Data were collected through semi-structured interviews conducted in person. The interviews followed a predefined interview guide containing open-ended questions related to teamwork experiences, conflict situations, communication challenges, and conflict-resolution strategies. However, the semi-structured format also allowed for follow-up questions to explore unexpected responses or clarify participant responses in greater detail.

Semi-structured interviews were considered particularly appropriate for this research because they enable participants to describe personal experiences in their own words while still maintaining enough structure to ensure consistency between interviews (Creswell, 2013), since the research focuses on perceptions of conflict and interpersonal dynamics. Therefore, the interview format allowed participants to discuss not only what conflicts occurred but also how they interpreted these situations emotionally and socially.

The semi-structured interview has taken inspiration from the Dringenberg and Purzer, 2018 interview guide, especially the introduction and logistical parts. The full interview guide is found in Appendix A. Questions have been structured and chosen so that personal bias is minimized, allowing the participant to provide rich and detailed responses about their experiences across four main areas: **(1) Context:** general context about the project; **(2) Perceiving Conflict:** conflicts that have occurred, their perception, and causes; **(3) Resolving Conflict:** how the conflict has been handled; and **(4) Outcomes:** how the participant reflects on the conflict in hindsight. If the participant is willing and sufficient time is left, the participant is asked about another project, and the same questions are posed in this context.

## 3.4 Data Analysis

The interview data were analyzed using Reflexive Thematic Analysis (RTA) as described by Braun and Clarke (2022). RTA was selected because it provides a flexible qualitative approach for identifying and interpreting patterns of meaning across participants' accounts while acknowledging the active role of the researcher in the analytic process. Since this study seeks to understand how students perceive and resolve conflict within collaborative software projects, RTA was considered suitable for exploring subjective experiences, interpretations, and reflections rather than measuring predefined variables.

The analysis followed the six-phase framework proposed by Braun and Clarke (2022). Familiarization with the data began during the local transcription of the audio recordings. The transcripts were then systematically analyzed in ATLAS.ti, where an inductive coding process was used to capture both explicit frictions and underlying dynamics. These initial codes were aggregated into candidate themes, which were reviewed against the dataset in stages to ensure it accurately reflected the students' perceptions. Finally, these themes were refined and named, resulting in the three primary themes.

# 4 Responsible Research

To ensure the integrity of this study, established ethical guidelines for qualitative research were adhered to throughout the data collection, analysis, and reporting phases (Creswell, 2013).

## 4.1 Participants

Prior to beginning the study, participants were contacted and informed of the general purpose of the research. Before the interview started, it was explained how the interview data would be utilized, and measures were taken to ensure that participants were not pressured into signing consent forms and that participation was strictly voluntary (Creswell, 2013).

During data collection, participants were respected by avoiding deception and ensuring they understood the scope of the interview. To maintain fairness and prevent exploitation, the use of leading questions was avoided, and the sharing of the researcher's personal opinions was actively withheld during the interviews (Creswell, 2013).

## 4.2 Researcher Positionality

As a final-year Computer Science student at TU Delft, the researcher possesses an "insider" perspective on the curriculum, the intensity of these collaborative software projects, and the common organizational frictions that occur. While this shared background was highly beneficial for building trust, conveying the extent of the study, and gaining access to participants, it also introduces potential researcher bias.

Having navigated the same academic environment, the researcher holds preconceived notions about the TU Delft Computer Science curriculum and may naturally sympathize with students who experienced severe institutional friction or active sabotage. To address this ethical issue and prevent personal experiences from compromising the study, specific measures were taken during the data analysis and reporting phases.

Specifically, active efforts were made to preserve researcher objectivity, ensuring that I did not side with participants simply due to relatable frustrations. By relying strictly on the Reflexive Thematic Analysis framework and utilizing ATLAS.ti to ground findings directly in the transcript data, multiple perspectives were reported objectively.

Furthermore, the analysis ensures the reporting of contrary findings and positive experiences, such as constructive conflict resolution and personal growth, rather than disclosing only the negative institutional results that might align with typical student grievances (Creswell, 2013).

## 4.3 Handling and Use of Sensitive Data

To ensure participant confidentiality and the secure handling of personally identifiable research data, strict data protection protocols were implemented. The semi-structured interviews were audio-recorded to capture participant speech accurately. To maintain data security, these recordings were transcribed locally using Whisper AI. After the transcription process, the data was thoroughly anonymized. Interviews were assigned a number, and any identifying information stated in the audio recordings was explicitly removed from the text, such as gender, name, and age.

All finalized transcripts and related survey materials are stored securely in a TU Delft-approved SURF-drive directory. Access to this secure directory is strictly restricted to the project researchers. Furthermore, any explicitly personally identifiable information (PII) collected, such as participant names, signatures, and phone numbers, was utilized solely for administrative purposes, including scheduling and the signing of the consent forms. This PII is completely excluded from all research outputs.

### 4.4 Use of AI

AI tools were utilized in a limited capacity. Generative AI was used to assist with structuring the written text of the report and as a brainstorming tool to help discover relevant papers. All AI-suggested literature was reviewed and verified. AI was not used to generate core academic arguments, conduct the RTA, or interpret the study's findings.

## 5 Results

Through reflexive thematic analysis of the interview transcripts, three primary themes have been identified that characterize how final-year Computer Science students perceive and resolve conflicts in collaborative software projects. The themes can be deconstructed by the conflict and outcome. To provide clarity on the resolution outcomes presented in the following themes, the terminology used to categorize the nature and impact of these resolutions is defined in Figure 1 and is further elaborated as follows:

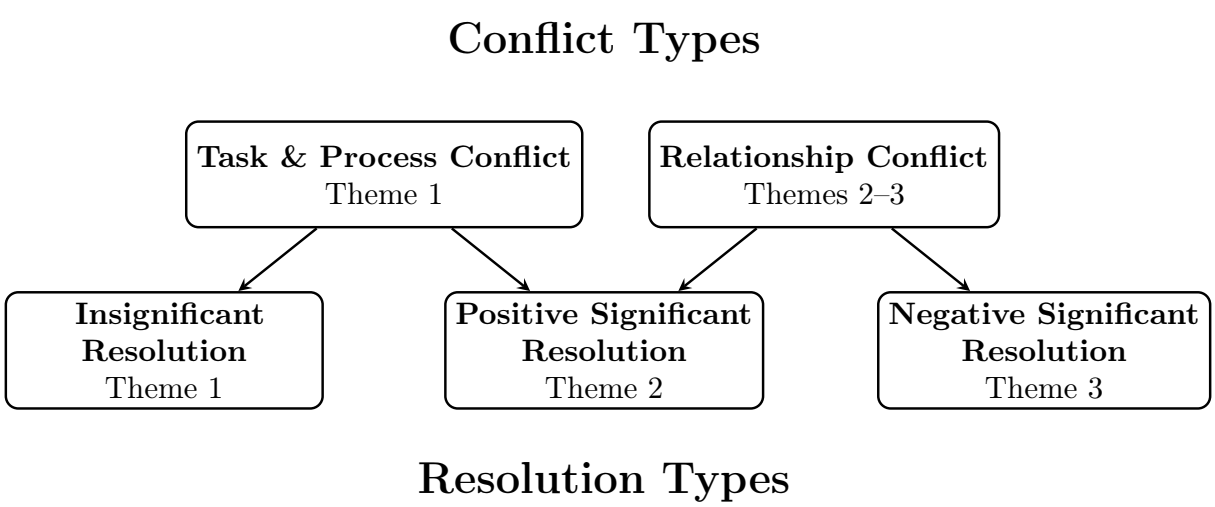


Figure 1: Conflict types and resolution outcomes aligned on a structured conceptual grid.

- **Insignificant** Pertains to minor, routine frictions, often strictly task-oriented, that are resolved quickly through standard team communication.
- **Significant** Refers to conflicts and their resolutions that have a lasting impact on the team's organization or the project's trajectory. These situations typically require sustained management, emotional labor, or external intervention.
- **Positive** Describes a resolution where the team or individual constructively adapts to the conflict. Even if the conflict itself was difficult, the resolution has a positive reflection.
- **Negative** Describes a resolution outcome characterized by terrible fallouts, where the student reflects negatively upon it's outcome.

### 5.1 Theme 1: Incidental and Constructive Conflict with Insignificant Resolution

This theme is described by the concept that conflict happens, but it is short, task-focused rather than personal, and does not negatively impact the team dynamic. It is resolved quickly through communication. It includes insignificant outcomes that are not perceived and reflected upon as either positively or negatively; thus, no positive or negative distinction is given to the theme. Table 1 shows the categories and codes associated with this theme. This theme is identified through the following categories after RTA:

Table 1: Codes associated with Theme 1: Incidental and Constructive Conflict with Insignificant Resolution.

| Category | Associated Codes | Description |
|---|---|---|
| **Task Frictions** | `Discussion about task`, `Task conflict`, `Focused roles` | Minor, inevitable disagreements regarding technical choices or task division. |
| **Positive Resolution** | `Goal or assignment was clear`, `Calm discussions`, `Decision-making based on opinion` | Quick neutralization of frictions through clear communication and shared expectations. |

**Task Frictions** This category captures the inevitable minor disagreements that arise regarding technical choices or task division. Codes such as *Task conflict* and *Discussion about task* illustrate that these frictions are focused on the work itself rather than on personal shortcomings. For instance, one student noted the collaborative nature of these frictions:

> **Participant A**: *"I wouldn't say it had unresolved conflict. To be fair we did have some discussion in the beginning on how we want to optimize it to work better. I expressed some concerns if we were to use a specific framework ... they proposed a solution to the problem ... I tried it, I liked it and, I said okay."*

**Positive Resolution** Frictions in this theme are quickly neutralized through clear communication and shared expectations. Captured by codes like *Calm discussions*, *Decision-making based on opinion*, and *Goal or assignment was clear*, the data shows that when foundational relationships are strong, teams can easily navigate task assignments without it bleeding into personal territory. One participant described this healthy dynamic:

> **Participant B**: *"No, we might have had some minor disagreements where we would just calmly talk it out as adults. Our arguments were very well-formed and well-backed ... we discussed them and we made decisions based on opinion. Stuff like that went very well."*

Another student highlighted how their willingness to discuss and compromise prevented friction entirely:

> **Participant D**: *"So we all had our tasks assigned and I did a version of my implementation. It worked... one particular group member, it was last five minutes and he wanted to force some changes. ... told the team about it and then we just basically confronted them or just told them to come so we can actually discuss about it. ... So at the end we kept his, because we agreed."*

Ultimately, the ability of these teams to utilize calm discussion and clear communication to quickly neutralize technical disagreements demonstrates that minor task conflicts can remain entirely constructive and incidental to the overall project.

## 5.2 Theme 2: Prolonged Behavioral Conflict with Positive, Significant Resolution

This theme is described by sustained behavioral issues managed constructively by the team and supervisors, ultimately leading to a formalized resolution and positive reflection. Table 2 shows the categories and codes associated with this theme. This theme is identified through the following categories after RTA:

**Teammate Absenteeism** This category involves chronic non-participation, highlighted by codes like *Absent team member*, *Teammate non-respondent*, and *Teammate misses meetings a lot*. Participants experienced ongoing frustration as they recognized the imbalance in workload. One participant noted:

Table 2: Codes associated with Theme 2: Prolonged Behavioral Conflict with Positive, Significant Resolution.

| Category | Associated Codes | Description |
|---|---|---|
| **Teammate Absenteeism** | `Absent team member`, `Teammate non-respondant`, `Misses meetings a lot`, `Communicates fickly` | Chronic non-participation and unreliability from a specific team member. |
| **Constructive Management** | `Accommodating to Problematic Teammate`, `Supervisor notices absent teammate`, `Problematic teammate suspended`, `Covering for teammate` | Professional, step-by-step attempts by the active team to resolve or work around the issue. |
| **Positive Growth** | `Skill development`, `Better scheduling after conflict`, `Leadership`, `Positive experience`, `Collaborative growth` | Personal and academic benefits realized after resolving the friction. |

> **Participant C**: *"I wouldn't say he was completely absent. He was a very special. I was a bit annoyed, especially for him not saying anything. Again, I think it's no problem if you don't know or if you can't, but at least you can mention it."*

**Constructive Management** Despite the frustration of absenteeism, the active team members attempt to resolve the issue professionally and with empathy. Codes such as *Accommodating to problematic teammate* reveal a progression from initially giving teammates the benefit of the doubt to pursuing formal intervention when the situation failed to improve. One participant described this process as follows:

> **Participant A**: *"We tried to understand whether if they have any external work or any like personal issues... we tried to be understanding towards them. Supervisors also said: 'okay this one person is not doing any work'... We tried approaching him a lot of times and ultimately he was suspended from our project."*

This account illustrates how teams initially sought to understand the underlying causes of a teammate's disengagement before escalating the issue to supervisors. When repeated attempts at resolution proved unsuccessful, formal intervention resulted in the teammate's removal from the project.

**Positive Growth** The resolution of the conflict leads to tangible personal and academic benefits. Characterized by codes like *Skill development*, *Better scheduling after conflict*, and *Leadership*, students reflect on the friction as a necessary learning curve. As one student realized:

> **Participant D**: *"I mean the conflict was pretty much resolved right after that and then afterwards there weren't really any issues so at the end the project I would say went smoothly."*

Ultimately, participants viewed struggle as beneficial to their professional development.

> **Participant A**: *"I managed to pick up a few skills about communication, about conflict resolution, and about leadership, so I think the project was a very positive experience for me."*

In the end, it was noted that they acquired valuable skills in communication, conflict resolution, and leadership. This progression from absenteeism to skill development confirms that teams can successfully adapt to and grow from prolonged behavioral conflicts when managed constructively.

### 5.3 Theme 3: Escalating Conflict and Significant Negative Outcome

This theme reflects participants' experiences of escalating team conflict, where problematic behavior extended beyond a lack of contribution and became disruptive to collaborative work. Participants further described instances in which formal support structures were perceived as insufficient, leaving teams to manage the consequences themselves. Table 3 shows the categories and codes associated with this theme. This theme is identified through the following categories after RTA:

Table 3: Codes associated with Theme 3: Escalating Conflict and Significant Negative Outcome.

| Category | Associated Codes | Description |
|---|---|---|
| **Disruptive team member behavior** | `Teammate sabotages work`, `Teammate barely does work`, `Interpersonal issue` | Participant accounts of behavior that extended beyond non-contribution and negatively affected team functioning and collaboration. |
| **Ineffective escalation processes** | `Escalation of concerns`, `Feeling frustrated`, `Seeking higher authority` | Experiences in which formal reporting and escalation procedures were perceived as providing limited resolution to ongoing team conflict. |
| **Significant Negative Consequences** | `Feeling let down`, `Study delay`, `Loss of trust`, `Nerve-wracking experience`, `Feeling self-doubt` | Reported academic, interpersonal, and emotional impacts associated with prolonged or unresolved conflict situations. |

**Disruptive Team Member Behavior** Participants described situations in which problematic teammate behavior extended beyond non-participation and was perceived as actively disrupting team processes. Captured by codes such as *Teammate sabotages work* and *Interpersonal issue*, these accounts reflected experiences where participants felt that a teammate's actions undermined collaborative work and created additional challenges for the team. A student described a situation like this as follows:

> **Participant A**: *"Later on when we had to put up the whole report we were waiting for one specific chapter of the report to show up and it doesn't show up... I think it involved all the other people in the project sending them an angry message that said: 'Hey, do your job or we're gonna report you.'"*

For participants describing these experiences, the conflict was no longer framed as a matter of unequal contribution. Instead, it was understood as behavior that interfered with both team collaboration and project progress.

**Ineffective Escalation Processes** Participants frequently described attempting to resolve these situations, sometimes even through formal channels. Codes including *Escalation of concerns*, *Seeking higher authority*, and *Feeling let down* captured experiences in which participants sought support from course staff, coordinators, or institutional bodies but perceived these efforts as having limited impact on the situation.

> **Participant B**: *"We went to the course coordinator about this... but he just blatantly said like hey I'm sorry but that's really not my problem... I actually made a case and took it to the board of examiners. I took it to the appeals board and it was rejected twice."*

These accounts suggest that escalation was often viewed as a final attempt to address an ongoing conflict. When participants perceived that formal procedures did not lead to meaningful intervention, feelings of frustration and disappointment frequently emerged.

**Significant Negative Consequences** Participants described a range of academic, interpersonal, and emotional consequences associated with prolonged conflict. Characterized by codes such as *Study delay*, *Loss of trust*, *Nerve-wracking experience*, and *Feeling self-doubt*, these accounts highlighted the lasting impact that unresolved team issues could have beyond the immediate project.

> **Participant B**: *"I now have a study delay of a full year due to this and the amount of stress it has given me... people have actually lost trust in me because of this... me not getting enough points definitely like confirmed their suspicions like hey maybe I'm not fit for this... the consequences were horrible."*

Ultimately, across these accounts, participants described how unresolved conflict could shape their academic progress, confidence, and relationships with others. Together, these experiences form a pattern in which escalating team conflict was associated with consequences that extended beyond the project itself and were perceived as having a lasting impact on participants' educational experiences.

# 6 Discussion

The themes identified through RTA of student interviews align closely with several established frameworks and previous findings within existing teamwork and conflict literature.

## 6.1 Comparison of Results

**Incidental Conflict and Task-Focused Disagreements** The task frictions and minor disagreements observed in Section 5.1 directly map to Jehn (1995)'s conceptualization of *task conflict*, which involves disagreements regarding ideas, decisions, or work content. Furthermore, the smooth and rapid resolution of these minor frictions supports the argument proposed by Deutsch that approaching disagreements cooperatively can lead to constructive outcomes without escalating interpersonal tension (Deutsch, 1973).

**Prolonged Conflict and Uneven Participation** The teammate absenteeism and chronic non participation detailed in Section 5.2 reflect the recurring student challenges identified by Iacob and Faily (2019), specifically the mismatches between teamwork expectations and the realities concerning workload distribution and accountability. Additionally, this behavior supports the findings of Dorić et al. (2025), who noted that undergraduate software engineering students frequently exhibit unevenly developed teamwork competencies, which can lead to coordination difficulties. Despite these challenges, participants' accounts of personal and academic growth through managing these situations support Dringenberg and Purzer (2018)'s assertion that engineering teamwork involves ongoing iteration, reasoning, and development throughout the project process. This growth extends beyond the technical product itself and includes the development of collaborative and interpersonal skills.

**Escalating Conflict and Significant Negative Consequences** The disruptive teammate behavior and escalation processes described in Section 5.3 can be interpreted as an intensified form of Jehn (1995)'s *relationship conflict*, which has been associated with reduced cohesion, trust, and collaboration effectiveness. Participants' accounts of academic setbacks, loss of trust, and emotional strain further align with the observations of Vanhanen et al. (2017), who found that teamwork problems can negatively influence both team satisfaction and perceived learning outcomes. These findings suggest that when conflict extends beyond disagreements about work and becomes embedded in interpersonal relationships, its consequences may reach beyond the immediate project experience.

## 6.2 Interpretation of Results

The findings illustrate how final-year BSc computer science students make sense of and navigate collaborative frictions within project-based software engineering education. The RTA indicates that participants generally demonstrate the capacity to address routine task-oriented disagreements and adapt constructively to structural challenges, such as uneven participation. Through these experiences, students often construct meaning

around conflict as a mechanism for developing interpersonal and leadership competencies, reinforcing the value of project-based learning as a practical environment for professional socialization and coordination.

However, the data also reveal patterns where interpersonal tensions become more complex and disrupt broader team cohesion. In instances of escalated conflict, students' accounts suggest a reliance on formal institutional support structures; yet, their subjective experiences indicate occasional difficulties in resolving these situations through standard reporting channels. In these cases, participants perceived the available institutional interventions as having a limited effect on their team dynamics and project outcomes.

Overall, this analysis highlights that while collaborative learning environments effectively support the development of technical skills, students' qualitative experiences of teamwork are linked to their perceptions of the available guidance and mediation. This points to the ongoing importance of ensuring students are adequately supported in utilizing conflict-resolution procedures. Rather than evaluating collaborative projects solely through final academic outcomes or deliverables. While tools like Buddycheck are currently utilized by the TU Delft BSc, all participants indicated that this was their first time reflecting on their teamwork. To move beyond evaluation metrics, courses could benefit from integrating alternative reflective practices. Such methods should foster genuine engagement, equipping students to actively identify, discuss, and manage interpersonal dynamics.

# 7 Conclusions

This study explored the qualitative experiences of final-year Computer Science students navigating and interpreting teamwork conflicts within collaborative software engineering projects. The findings demonstrate that students generally possess the capacity to constructively manage task-oriented frictions by relying on clear communication and the establishment of shared expectations. Furthermore, the RTA highlights that students often conceptualize prolonged behavioral challenges, such as uneven participation, as practical opportunities to foster essential professional competencies. While participants consistently demonstrated resilience and adaptability in their conflict resolution strategies, the research also emphasizes that navigating complex relationship conflicts requires accessible institutional support structures to prevent interpersonal tensions from impacting the educational experience and project outcomes.

However, the study has some limitations. Constrained by a ten-week time-frame, the study relied on a small sample of four students recruited via personal academic networks, which may limit the generalizability of the findings. Additionally, the researcher's status as a final-year TU Delft student provides a valuable insider perspective but may introduce potential bias when interpreting peer experiences.

Therefore, future studies should utilize a longer timeframe and a larger, more diverse participant pool to capture a wider array of teamwork dynamics. On the education side, Software engineering programs could embed dedicated conflict management training within collaborative coursework. Providing students with structured strategies to address and resolve team disputes will transform inevitable friction into an active learning opportunity. Finally, universities may benefit from evaluating and refining their formal support and mediation channels. Ensuring accessible and transparent mechanisms for conflict resolution provides students with appropriate guidance when complex interpersonal dynamics exceed the team's capacity for internal management.

Ultimately, collaborative projects serve as an invaluable training ground for future software engineers. While interpersonal challenges are inevitable, they provide opportunities for students to develop essential communication and leadership skills. By transforming teamwork obstacles into personal growth, these graduates secretly gain the resilience and conflict resolution capabilities needed to thrive as effective professionals in the software industry.

# References


Braun, V., & Clarke, V. (2022). *Thematic Analysis: A Practical Guide*. SAGE Publications.

Cohen, S. G., & Bailey, D. E. (1997). What Makes Teams Work: Group Effectiveness Research from the Shop Floor to the Executive Suite. *Journal of Management*, *23*(3), 239–290. https://doi.org/10.1177/014920639702300303

Creswell, J. W. (2013). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (4th). SAGE Publications.

Deutsch, M. (1973). *The Resolution of Conflict, Constructive and Destructive Processes* (Vol. 17). American Behavioral Scientist. https://doi.org/10.1177/000276427301700206

Dorić, L., Luburić, N., Slivka, J., Prokić, S., & Đukić, A. (2025). Evaluating Teamwork Components in Large Undergraduate Software Engineering Teams. *ACM Transactions on Computing Education*, *25*(3). https://doi.org/10.1145/3733840

Dringenberg, E., & Purzer, Ş. (2018). Experiences of First-Year Engineering Students Working on Ill-Structured Problems in Teams. *Journal of Engineering Education*, *107*(3), 1–26. https://doi.org/10.1002/jee.20220

Iacob, C., & Faily, S. (2019). Exploring the gap between the student expectations and the reality of teamwork in undergraduate software engineering group projects. *Journal of Systems and Software*, *157*, 110393. https://doi.org/10.1016/j.jss.2019.110393

Jehn, K. A. (1995). A Multimethod Examination of the Benefits and Detriments of Intragroup Conflict. *Administrative Science Quarterly*, *40*(2), 256–282. https://doi.org/10.2307/2393638

Katzenbach, J. R., & Smith, D. K. (1993). The discipline of teams. *Harvard Business Review*, *71*(2), 111–120. https://pubmed.ncbi.nlm.nih.gov/10124632/

Kimpton, C., & Maynard, N. (2025). Factors shaping teamwork skills development in tertiary engineering education: a systematic literature review. *European Journal of Engineering Education*, *50*(2), 253–280. https://doi.org/10.1080/03043797.2024.2357343

Kozlowski, S. W. J., & Ilgen, D. R. (2006). Enhancing the Effectiveness of Work Groups and Teams. *Psychological Science in the Public Interest*, *7*(3), 77–124. https://doi.org/10.1111/j.1529-1006.2006.00030.x

Marques, M., & Ochoa, S. F. (2014). Improving Teamwork in Students Software Projects. *Proceedings of the 27th IEEE Conference on Software Engineering Education and Training (CSEET)*. https://doi.org/10.1109/CSEET.2014.6816787

Thomas, K. W. (1992). Conflict and Conflict Management: Reflections and Update. *Journal of Organizational Behavior*, *13*(3), 265–274. https://doi.org/10.1002/job.4030130307

Vanhanen, J., Lehtinen, T. O. A., & Lassenius, C. (2017). Software engineering problems and their relationship to perceived learning and customer satisfaction on a software capstone project. *Journal of Systems and Software*. https://www.sciencedirect.com/science/article/pii/S0164121217302716

Wu, Y., Xu, L., & Philbin, S. P. (2023). Evaluating the Role of the Communication Skills of Engineering Students on Employability According to the Outcome-Based Education (OBE) Theory. *Sustainability*, *15*(12), 9711. https://doi.org/10.3390/su15129711

# A Appendix: Full Interview Guide

The full interview guide used in the semi-structured interviews is found below.

## Logistics

- Thank you for volunteering for this interview, but before we get into the actual interview, I want to let you know what you can expect.
- Our conversation will be recorded and transcribed. This will allow me to look back at our conversation, such that I don't continuously have to be writing notes. Any information which could identify you, will be removed and the audio will be stored securely on a private SURFdrive directory hosted by the TU Delft.
- Your name will be replaced by a random pseudonym.
- Sometimes I will make notes, so I can ask follow up questions later, without interrupting you.
- The interview is about an hour long, but it is voluntary. You can leave anytime you want.
- Do you have any other questions before we start?

*** Ensure the consent form is signed ***
*Start recording*

- The purpose of this interview is how you experienced teamwork within collaborative software projects.
- I want to hear your own experiences, and how you perceived them. There is no right or wrong answer.
- I want to understand your perspective, so I will ask questions like "what do you mean by that?" These types of questions can be a bit awkward and may require you to reflect on what you have experienced, so I will purposefully leave long pauses after my questions.
- Please take your time, if you need clarification, please let me know.

## Context

- To start, could you tell me about a recent collaborative software project you worked on, that stands out in your memory?"
  - What was the project about, and how large was the team?
  - How did your team initially organize the work and assign roles?

## Perceiving Conflict

- Collaboration rarely goes perfectly smoothly. Can you walk me through a specific time during this project when the team or you disagreed or experienced friction?
  - What was the cause of the disagreement / friction?
  - At what point did you realize that this was an actual conflict rather than just a standard discussion?
  - How did this realization make you feel?
  - What type of problem did you perceive it as?
  - Did you team have many conflicts? Why or why not?

## Resolving Conflict

- Once the conflict was apparent, walk me through what happened next. How did the team attempt to handle it?
  - Who initiated the resolution process?
  - What was your specific role in addressing or navigating this situation?
  - Were there any specific strategies or tools your team used to get past the disagreement?
  - (Optional) Did you involve instructors or TAs? Why or why not?

## Outcomes

- What was the ultimate outcome of that specific conflict? How did it impact the final result?
  - Looking back at how the situation was resolved, how do you feel about the outcome now?
  - What, if anything, would you have done differently?
- Has this specific experience changed the way you plan to approach teamwork in your future career as a computer scientist?
  - How have your perceptions of teamwork changed?
  - What qualities about you, perhaps some way you are different from your peers, influence your experience in teamwork projects?
- Is there anything else you would like to share to help me understand your experience?

## Conclusion

- What questions do you have for me?
- Do you have any feedback for me that would improve the experience of the next participant?