Exploring Assessment Approaches for Student Group Projects in Undergraduate Computing Education

MICHAEL JAMES SCOTT, Falmouth University, United Kingdom
RITA GARCIA, Victoria University of Wellington, New Zealand
ELLIE LOVELLETTE, College of Charleston, United States
SEÁN RUSSELL, University College Dublin, Ireland
XI WU, University of Sydney, Australia
BEDOUR ALSHAIGY, Uppsala University, Sweden
HIEKE KEUNING, Utrecht University, The Netherlands
SANDRA SCHULZ, University of Potsdam, Germany
ALEXANDER MITCHELL, Falmouth University, United Kingdom
MIRELA GUTICA, British Columbia Institute of Technology, Canada
P D PARTHASARATHY, BITS Pilani, India

Teamwork and group projects are a prominent feature of undergraduate computing education. At the tertiary level, educators use group projects to nurture professional skills and employability. Yet, student concerns about fairness continue to drive discourse around what is assessed and how. Instructors may choose to evaluate—either individually or collectively—principles, the process, the final product, professional behaviour, or students' progress. However, the range of methods for assessing student group projects in undergraduate computing remains insufficiently characterised. This article takes a two-pronged mixed-methods approach, surveying assessment approaches and interviewing educators. A Systematic Literature Review (SLR) of studies published from 2013 to 2024 that examine the assessment of student group projects in undergraduate computing contexts identifies considerable variation in assessment practices, which are synthesised through qualitative content analysis into a schema of 23 distinct modes. A series of eighteen long-form semi-structured interviews with instructors of group project-based computing courses from around the globe examines instructor perspectives and challenges. Concerns about the parity and inclusivity of some methods permeate the literature and are echoed in the interview findings. Peer evaluation is a recurring theme, as are tools designed to monitor individual contributions to shared deliverables. While there is limited empirical evidence supporting the suitability of any one approach, there are also intriguing opportunities which computing departments are well-placed to implement, such as versioning, tracking, and analytics. The article concludes by listing recommendations for assessment practice and outlining directions for future research to improve the comparability, transparency, and effectiveness of group project assessment in undergraduate computing education.

CCS Concepts: • **Social and professional topics → Computing education programs**; *Model curricula.*

Additional Key Words and Phrases: assessment, student teams, group projects, computing education, university

Authors' Contact Information: Michael James Scott, Falmouth University, Penryn, Cornwall, United Kingdom, michael.scott@falmouth.ac.uk; Rita Garcia, Victoria University of Wellington, Victoria, New Zealand, rita.garcia@vuw.ac.nz; Ellie Lovellette, College of Charleston, Charleston, South Carolina, United States, lovelletteeb@cofc.edu; Seán Russell, University College Dublin, Dublin, Ireland, sean.russell@ucd.ie; Xi Wu, University of Sydney, Sydney, Australia, xi.wu@sydney.edu.au; Bedour Alshaigy, Uppsala University, Uppsala, Sweden, bedour.alshaigy@it.uu.se; Hieke Keuning, Utrecht University, Utrecht, The Netherlands, h.w.keuning@uu.nl; Sandra Schulz, University of Potsdam, Potsdam, Germany, sandra.schulz@uni-potsdam.de; Alexander Mitchell, Falmouth University, Penryn, Cornwall, United Kingdom, a.mitchell@falmouth.ac.uk; Mirela Gutica, British Columbia Institute of Technology, Burnaby, BC, Canada, Mirela_Gutica@bcit.ca; P D Parthasarathy, BITS Pilani, KK Birla Goa Campus, India, p20210042@goa.bits-pilani.ac.in.

1 Introduction

Effective collaboration is essential in the computing sector, where developing complex software requires seamless teamwork among diverse team members [Nwulu et al. 2023], particularly in increasingly distributed contexts that need to be as agile as possible [Rizvi et al. 2015]. Employers consistently prioritise interpersonal, teamwork, and communication skills alongside technical proficiency when evaluating computing professionals [Groeneveld et al. 2019]. As such, the ability to work productively within a team, negotiate decisions, and convey technical information is valued just as much as proficiency in programming languages or algorithm design. Recognising this, The Joint ACM/IEEE/AAAI Task Force on Computing Curricula [2023] recommends that undergraduate computing curricula incorporate opportunities for students to cultivate professional skills.

Various strategies exist for integrating collaborative skill development into computing programmes [Kirk et al. 2022]. Among these are group projects, which require self-managing student teams to deliver a shared solution to a non-trivial problem over an extended period. Typically, these projects serve as complex environments where students navigate social and technical processes [Wiggberg 2008]. Recognising this dual nature, educators have long treated group work as a cornerstone of the computing curriculum [Fincher et al. 2001].

A recent systematic mapping study by Kokkoniemi and Isomöttönen [2023] explores the considerable body of literature on student group projects in software engineering. They found that many studies did not explicitly engage with collaborative learning theories and empirical validation of teaching interventions is limited. These findings align with student perspectives: while many students acknowledge that group projects are valuable learning experiences, they also identify considerable scope for improvement [Licorish et al. 2022; McKay and Sridharan 2024]. Although a large number of studies focus on students' experiences, Hooshangi et al. [2025] explored instructors' perspectives. They found that team-based capstone course structures tend to share more commonalities than differences, though recurring considerations emerged including team formation, managing student conflicts, and equitable grading practices.

Assessment remains a persistent and structurally complex challenge with current methods struggling to reflect the multifaceted nature of teamwork [Hooshangi et al. 2026; Mitchell et al. 2024; Scott et al. 2023]. Yet, studies into the assessment process are relatively under-represented [Kokkoniemi and Isomöttönen 2023]. This has led to persistent concerns about the fairness, validity, and reliability of assessment practices [Gordon 2010]. This friction is apparent in the unresolved tension between evaluating collective outputs and individual contributions [Richards 2009], a division that often manifests as student frustration over unequal contribution or unrecognised effort [Hooshangi et al. 2026; Polack-Wahl 2001; Porquet-Lupine and Brigham 2023]. Ultimately, some educators find designing an assessment strategy that accurately captures nuanced team dynamics whilst simultaneously nurturing an equitable, transparent learning environment to be an elusive goal [Tafliovich et al. 2015].

These challenges suggest that difficulties in assessing student group projects are not merely practical, but stem from the inherently multidimensional nature of teamwork and its evaluation. As a result, there is value in abstracting assessment practices into more structured representations that allow their key components and relationships to be examined systematically. Vasilevskaya et al. [2015] introduced modelling of assessment design in the context of group projects, characterising assessment as a design artefact comprising structural elements and relationships that describe how assessment operates. While teaching contexts and assessment purposes vary across institutions, recurring patterns can still be observed, such as alignment between intended learning outcomes and the tasks used to elicit evidence. This supports the view that assessment can be modelled at a conceptual level. Lopez et al. [2025], for example, demonstrates this through a model of group capstone assessment in computer engineering. Such models provide a means to formalise shared design logics, enabling researchers and practitioners to reason about assessment approaches in a structured and comparative way and to support more principled design of student group project assessment.

While prior reviews address peer assessment [Yang et al. 2022] and group-based assessment [Tumpa et al. 2022] in broader educational settings, and contemporaneous work explores assessment in team-based capstone contexts [Hooshangi et al. 2026], there remains a lack of dedicated synthesis regarding group project assessment in undergraduate computing education across both capstone and non-capstone curricula. Many studies report isolated implementations of assessment practices with limited theoretical integration or cross-context synthesis [Kokkoniemi and Isomöttönen 2023]. Representative examples of commonly recurring but context-specific implementations include peer assessment and moderation systems [Babo et al. 2020], rubric-based teamwork evaluation [Vaughan et al. 2019], and contribution tracking for individual weighting mechanisms [Buffardi 2020]. We therefore argue that the field would benefit from stronger theoretical integration across contexts to inform the design of student group project assessment. By organising contemporary approaches described across the literature, this article takes a step towards clarifying the range of assessment strategies in use by offering a structured conceptual framework of these methodologies, exploring the challenges associated with their implementation, and illustrating technologies that can support complex or otherwise unwieldy assessment processes.

Accordingly, this article presents a systematic review of the literature supported by findings from interviews with computing instructors on approaches to assessing student group projects in undergraduate computing, striving to address the following research questions:

RQ1: What assessment approaches are typical of student group projects in undergraduate computing education?

RQ2: What challenges do undergraduate computing educators encounter when assessing student group projects?

RQ3: How is technology used to support the assessment of student group projects in undergraduate computing?

The remainder of this article is structured as follows. Section 2 reviews related work on group projects and assessment, giving further background whilst also clarifying relevant concepts and terminology. Section 3 describes the methodology used in this work. It spells out the Systematic Literature Review (SLR) protocol and provides insight into the content analysis procedure followed to derive data from each paper. It also outlines the semi-structured interview protocol and the approach used for the subsequent data analysis. Section 4 presents the SLR findings, organised around the research questions. Section 5 discusses the results of the qualitative analysis of the interviews. Section 6 synthesises insights into the different methods of assessment, discussing implications for practice. Section 7 provides practical recommendations from our work, while Section 8 presents the limitations for this study. Finally, Section 9 summarises the findings and outlines directions for future research.

2 Background

This section establishes the conceptual and theoretical foundations for the study. Section 2.1 defines student group projects within computing education and situates them within broader curricular and professional contexts. Section 2.2 examines the pedagogical and assessment challenges associated with collaborative project work, including tensions surrounding fairness, accountability, and multidimensional learning outcomes. Finally, Section 2.3 introduces assessment approaches as an analytical framework for examining how collaborative assessment practices are structured and operationalised.

2.1 Student Group Projects in Computing

Student group projects have been a longstanding feature of computing education [Fincher et al. 2001]. Early curriculum recommendations and subsequent reports advocated for collaborative project work to help students develop both technical and professional competencies [Atchison et al. 1968; Binns and West 1975]. Over time, it has become an integral part of the design of computing curricula [Wiggberg 2008], reflecting broader shifts in educational goals and growing recognition of teamwork as an essential graduate attribute [English and Hayes 2022]. This emphasis on employability skills through group projects aligns with broader discussions about the relationship between computing curricula and industry needs. Garousi et al. [2019] conducted a meta-analysis of curricular reforms, identifying deficiencies in areas such as configuration management and software testing that impact graduates' professional readiness. Similar concerns have been raised in programming education [Luxton-Reilly 2009], and in subfields such as game development, where teamwork cultures can exacerbate workload distribution and coordination problems [Harvey 2019]. More recent curriculum guidance produced by The Joint ACM/IEEE/AAAI Task Force on Computing Curricula [2023] explicitly endorses teamwork as an essential component of computing education and professional preparation.

In undergraduate computing curricula, collaborative projects are integrated across diverse pedagogical contexts. These range from fundamental programming instruction [Blake 2005; Scott et al. 2015] to comprehensive capstone experiences that incorporate external stakeholder engagement [Bruegge et al. 2015; Steghöfer et al. 2018; Stephenson et al. 2016], interdisciplinary collaboration [Gunnars and Palmquist 2026; Paredes-Velasco et al. 2023], and the adoption of authentic practices from, for example, free and open source (FOSS) projects [Braught et al. 2018]. While such initiatives are frequently situated within the final stages of a degree programme [Bruegge et al. 2015; Whalley et al. 2025], Bernhart et al. [2006] provides a framework for term-long team projects which can be used earlier in programme delivery to scaffold the development of software engineering competencies.

There are many reasons for students to group together, and so the computing education literature sometimes uses the term student group project imprecisely, blurring the boundary between collaborative learning and formal projects [Kokkoniemi and Isomöttönen 2023]. As such, it is important to distinguish informal, unstructured collaborative tasks [Mason 2020] from the explicit instruction and formal structures that scaffold professional skill development [Lingard and Barkataki 2011]. Expanding upon the work of Davies [2009], we define *student group project* as a practical assignment in an educational setting where at least three students work collaboratively to produce a shared deliverable, on the basis of their own initiative, over an extended period. Team-based capstone projects represent a common manifestation of this type of project in computing education [Whalley et al. 2025]; however, our definition is broader and includes comparable forms of collaborative project work across the curriculum.

This article operationalises student group projects in computing using five criteria:

(i) **Artifact-Centric:** The activity produces a substantive computing-related artefact or solution as a shared outcome. This criterion reflects the project-based learning strategy [Krumova et al. 2020]. Collaborations oriented toward applied competence development primarily from conceptual discussion or shared inquiry are excluded;

(ii) **Integrative Collaboration:** The project requires students to work "as a group" to contribute collectively to a unitary deliverable [Chiriac 2014]. This creates a conjunctive task structure [Steiner 1972] where contributions to the deliverable integrate in a sophisticated manner, necessitating positive interdependence [Johnson and Johnson 1999] and usually demands the application of project management techniques to coordinate. Thereby distinguishing it from additive tasks "in a group," where individual contributions are merely aggregated;

(iii) **Polyadic Interaction:** At least three students must work together. The dynamic that emerges in groups of this size and larger necessitates negotiation and mediation [Simmel 2009], requiring interpersonal management [Burke 2011]. This excludes pair programming which is a dyadic interaction rather than a polyadic interaction, often focusing on synchronous problem-solving where complex coordination is unnecessary [Lasserre and Szostak 2011];

(iv) **Complex Requirements:** The project requires a "non-trivial" and "ill-defined" objective, typically drawing on resources no single individual possesses [Cohen 1994]. Such projects challenge student agency and creativity, establishing a shared aim for the group [Davies 2009] that necessitates diverse perspectives and collective problem-solving. Consequently, excluding rote adherence to instructions;

(v) **Persistence:** The group maintains stable membership over an extended period [Davis 1993], typically several weeks to a full study block or perhaps even several consecutive study blocks. This time-span fosters the cohesiveness [Davies 2009] and shared history required for a group to move beyond a collection of individuals into a performing team, facilitating a full lifecycle of team development, from forming to performing [Tuckman 1965]. Thereby distinguishing it from short-term or ad-hoc in-class activities.

Taken together, these characteristics distinguish student group projects from short-lived or loosely coordinated collaborative activities while highlighting the socially and technically complex contexts within which assessment judgements are made.

2.2 Assessment Challenges in Student Group Projects

Group projects are widely used within computing curricula because they provide opportunities for students to develop technical, collaborative, and professional skills in contexts intended to resemble professional software development [Bastarrica et al. 2017; Tenhunen et al. 2023]. Consequently, assessment in such projects must account not only for technical outcomes, but also for teamwork practices, communication, and individual contribution.

Studies of student software teams have additionally demonstrated coordination problems comparable to those observed in professional software engineering contexts [MacKellar 2013]. Without explicit training or structured support, student teams often function on an ad hoc basis, resulting in ineffective collaboration and unresolved tensions [Lingard and Barkataki 2011; Scott et al. 2019].

Several assessment theorists have emphasised the importance of aligning assessment design with intended learning processes and outcomes. Biggs's [1996] theory of constructive alignment, alongside subsequent work on formative assessment and feedback [Gibbs 1988; Nicol and Macfarlane-Dick 2006], argues that learning outcomes, teaching activities, assessment tasks, and feedback processes should operate coherently to promote deep and self-regulated learning. Within group projects, however, these principles are difficult to operationalise because assessment must simultaneously capture technical achievement, collaborative practice, and individual contribution.

Traditional assessment approaches often struggle to capture the complexity of collaborative work and to fairly allocate marks to individuals for their contributions [Gordon 2010]. Students regularly express frustration when unequal effort is not appropriately recognised [Polack-Wahl 2001; Porquet-Lupine and Brigham 2023]. Assessment complexity in student group projects arises not only from collaboration itself, but from the need to evaluate multiple forms of learning simultaneously. A central difficulty is that group projects require educators to assess learning that is simultaneously collective and individual in nature.

These collaborative demands create persistent practical challenges for students, including scheduling difficulties, unequal contributions, interpersonal conflict, and the coordination of shared resources [Dorić et al. 2023; Polack-Wahl 2001; Porquet-Lupine and Brigham 2023]. In response, a range of pedagogical interventions have been proposed to scaffold teamwork skills and improve collaborative practice [Kimpton and Maynard 2025]. For example, Mahmood [2007] recommend structured approaches that balance technical and interpersonal skill development within project-based learning, while Peters et al. [2019] describe learning agreements that encourage accountability, goal articulation, and reflective practice within teams.

Peer assessment is widely proposed as a mechanism for fairness and accountability, as it fosters active engagement and critical reflection through a constructivist lens [Topping 1998]. Yet concerns persist regarding its reliability, social pressure, and student trust [Mitchell et al. 2021, 2022]. Although early studies suggested peer marks could correlate with teacher grades [Falchikov 1995], subsequent research indicates that students often inflate marks due to social pressures [Isomöttönen 2014] and express skepticism regarding the fairness of summative peer evaluations [Kay 2022]. To mitigate these challenges, empirical evidence underscores that structured training enhances rating accuracy, fosters better feedback, and increases student confidence in the fairness of the process [Liu and Li 2014; Stonewall et al. 2018]. Careful design and training are especially important when such assessments affect final grades [Hanrahan and Isaacs 2001]. Fagerholm and Vihavainen [2013], for example, applied Bloom's revised taxonomy to structure peer feedback activities, promoting higher-order cognitive processes such as analysis, evaluation, and creation.

Learning outcomes describe what learners are expected to know, understand, and apply following a learning activity [Biggs 2003]. Within student group projects, assessment may target a broad range of technical, interpersonal, and professional capabilities. Bloom's revised taxonomy [Krathwohl 2002] provides a useful framework for articulating such outcomes by distinguishing between cognitive processes and knowledge dimensions. In the context of collaborative computing projects, however, assessment must additionally account for forms of learning that extend beyond discrete technical competencies, including teamwork practices, professional conduct, and reflective development.

Building on the framework proposed by Clear [2009], Herbert [2018] categorise learning within computing group projects according to four interrelated dimensions: *Product*, *Process*, *Professionalism*, and *Progress*. Together, shown in Table 1, these categories capture both the technical outputs of collaborative work and the developmental and interpersonal dimensions of student learning. Because Clear [2009] primarily focused on capstone contexts, where disciplinary competence may be assumed to some extent, this article extends the framework by introducing a fifth dimension: *Principles*. This additional category captures students' demonstration and application of foundational disciplinary knowledge and technical concepts, which may be particularly important in earlier stages of undergraduate computing education.

Taken together, these dimensions illustrate why assessment in group projects is inherently multidimensional: educators must simultaneously evaluate technical competence, collaborative practice, professional behaviour, and individual learning trajectories within a shared project context. As Hans [2024] observe, the relative emphasis placed on these dimensions depends on how project success is conceptualised, with important implications for grading practices and student perceptions of fairness. This has contributed to longstanding debates regarding whether achievement should be attributed primarily to groups or differentiated at the level of individual contribution [Richards 2009].

Foundational work by Sluijsmans et al. [2001] and subsequent meta-analyses by Van Zundert et al. [2010] suggest that differentiating individual performance is important for perceived fairness, although implementation remains highly context-dependent. These challenges remain evident

Table 1. The 5Ps for Student Group Project Assessment

Assessment Dimension	Description
Product	The technical artefact or deliverable produced through collaborative effort, representing the application of disciplinary knowledge and practical skills
Process	The collaborative and organisational practices through which work is undertaken, including communication, coordination, participation, and project management activities
Professionalism	The conduct and disposition of individuals and teams as they engage with responsibilities, collaborators, and constraints in accordance with disciplinary and ethical expectations
Progress	The development of learner capability over time, typically demonstrated through reflection on personal growth, challenges encountered, and future improvement
Principles	The demonstration and application of foundational disciplinary knowledge, theories, and technical concepts within the project context.

within contemporary analyses of team-based capstone assessment [Dugan Jr 2011; Hooshangi et al. 2026; Tenhunen et al. 2023; Whalley et al. 2025]. More recent work has therefore explored holistic heuristics for allocating marks to individual group members [Russell 2025].

Collectively, this literature demonstrates that assessment in student group projects extends beyond evaluating technical artefacts alone. Effective assessment strategies must additionally account for collaborative processes, individual participation, professional conduct, and learner development, while also addressing the evidentiary challenges associated with attributing achievement within collective work. These competing demands create persistent tensions between fairness, validity, individual accountability, and pedagogical coherence. As such, assessment design within computing group projects remains an ongoing area of pedagogical and research concern.

2.3 Assessment Approaches

An assessment approach is a structured and abstract representation of how assessment practices within a learning environment are designed to elicit, interpret, and act upon evidence of student attainment. Rooted in the principles of evidence-centred design (ECD) [Mislevy et al. 2003a,b] and constructive alignment [Biggs 2003], an assessment approach defines the conceptual blueprint that guides the coherent integration of pedagogical aims with task design, evidentiary reasoning, and delivery mechanisms [Mislevy and Riconscente 2006]. The assessment approach is distinct from a specific instance of assessment activity, instead functioning as a high-level abstraction that is formally decoupled from the technical and contextual specificities of any single implementation [Mislevy et al. 2003a]. By maintaining this separation of concerns, the approach provides a reusable schema that ensures the internal logic-linking pedagogical goals to evidentiary reasoning remains conceptually coherent even as it is adapted into diverse assessment activities and delivery formats [Mislevy and Riconscente 2006]. This structural coherence is critical; Fink [2013] posits that misalignment between learning outcomes, teaching strategies, and assessment constitutes a ‘broken’ design. A well-defined approach ensures integrated alignment by generalizing these

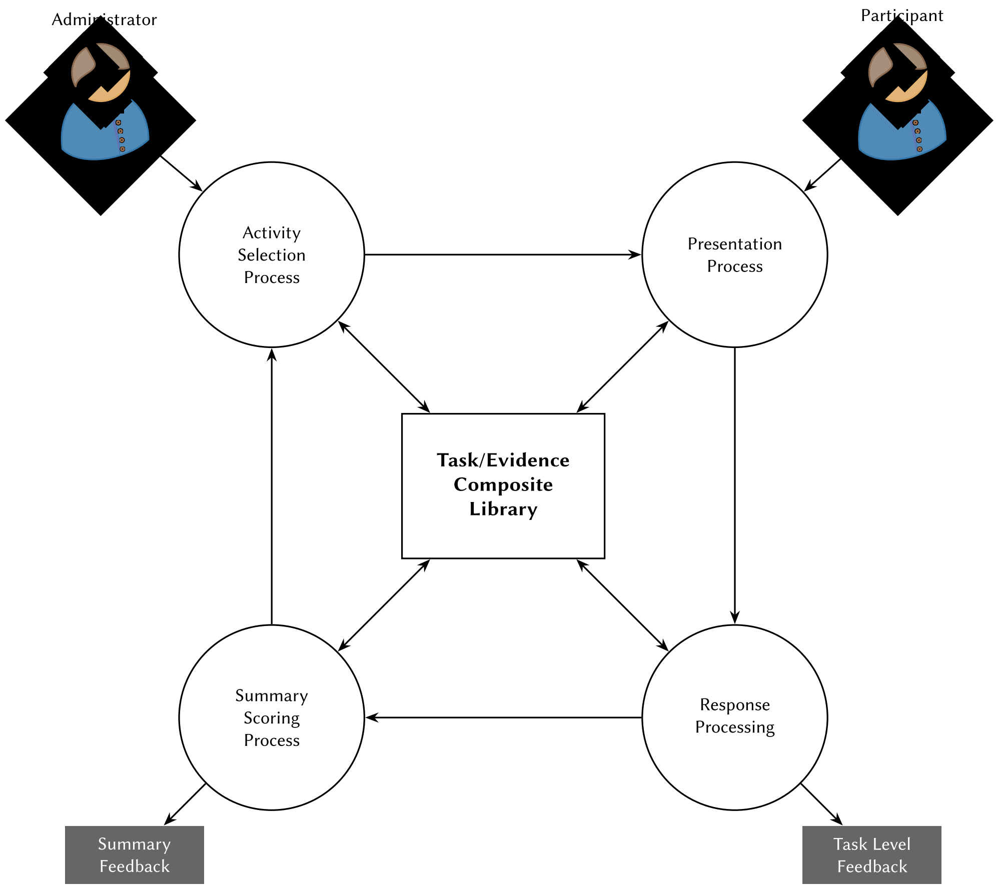


Fig. 1. Almond's framework — a four process architecture for assessment [Almond et al. 2002]

relationships across contexts to support robust reuse and adaptation [Brown and Knight 1994; Race 1999].

Almond et al. [2002] provide a conceptual framework for describing an assessment, shown in Figure 1, which we use as inspiration for our study. Although Almond's framework has been situated to inform the design of e-learning systems, it is rooted in universal principles of evidentiary reasoning and thus is useful for operationalising our notion of an assessment approach. We use these concepts as a framework for describing group assessment. It incorporates key components that ensure assessments are systematically designed and effectively implemented. The framework supports translating learning goals into measurable learning outcomes.

Alternative frameworks such as the *BEAR Assessment System* [Wilson and Sloane 2000] and the *Assessment Design Decisions Framework* (ADDF) [Bearman et al. 2016] address developmental progress and high-level pedagogical choices. The *BEAR Assessment System* establishes developmental "construct maps" to ensure curriculum alignment, yet it primarily focuses on the *why* and *what* of assessment within a classroom ecosystem. Similarly, the *ADDF* provides a comprehensive guide for high-level decision-making in a way that balances tensions between purpose, feedback, and student interactions. However, it does not provide the granularity required to map how raw observations are synthesised to measure attainment. In contrast, Almond et al.'s [2002] four-process architecture (4PA) is specifically designed to operationalise the transformation of raw student performance into evaluative judgements. This decomposition provides a rigorous basis for analysing information flow in collaborative assessment, where the relationship between individual contribution and awarded grade is often opaque. Although 4PA primarily emphasises structural relationships rather than pedagogical interpretation, it nevertheless offers a systematic means of distinguishing how tasks are presented, how students respond, and how those responses are transformed into evaluative outcomes. The framework therefore makes explicit the evidentiary logic underlying assessment: namely, how group interactions are captured, interpreted, and aggregated as individual or collective evidence of attainment. Consequently, 4PA enables granular comparison between complex assessment approaches and provides the level of structural transparency required for this study.

Table 2 explains the role of each of the four processes:

Table 2. The role of each process in Almond et al.'s [2002] four process architecture

Process	**Role**
Activity Selection	Selecting and sequencing tasks intended to elicit evidence of learning
Presentation	Delivering tasks to learners and capturing their responses
Response Processing	Interpreting responses and generating task-level observations or feedback
Summary Scoring	Aggregating observations to produce overall evaluative judgements

Operationally, these four processes are realised through six interrelated components within the broader assessment framework, as illustrated in Figure 2 and expanded in Table 3 below:

Table 3. The role of each model in Almond et al.'s [2002] four process architecture

Model	**Role**
Student	Specifying the constructs, knowledge, skills, and dispositions to be inferred through assessment.
Task	Characterising the features and formats of tasks designed to elicit evidence of those outcomes.
Evidence	Defining how task performances are interpreted as observations and how those observations support inferences about learner attainment.
Assembly	Governing the selection, sequencing, and combination of tasks into a coherent assessment instrument.
Presentation	Defining how tasks and instructions are communicated to participants and how responses are collected.
Delivery	Specifying the administrative, technical, and logistical conditions under which assessment is conducted.

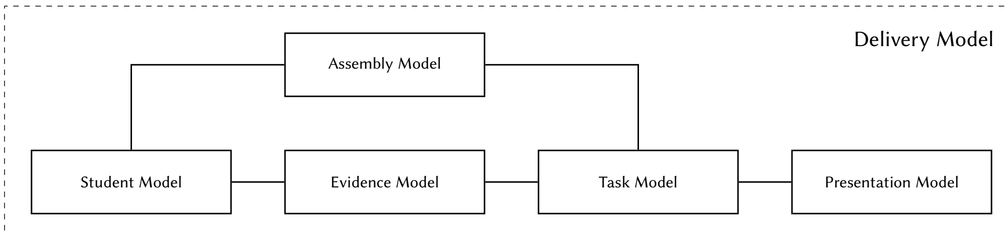


Fig. 2. Almond et al.'s [2002] principle design components of the assessment framework

Taken together, the four processes and six components provide a coherent operationalisation of assessment practice. The framework functions both as a design-oriented planning instrument and as a conceptual structure for analysing how assessment evidence is generated, interpreted, and reported. For the purposes of collaborative assessment, the distinction between the *Task*, *Evidence*, and *Student* models is particularly significant. Collaborative tasks frequently generate intertwined evidence of collective and individual performance, requiring explicit reasoning about how observations are attributed, interpreted, and aggregated into evaluative judgements. Almond's framework provides a structured means of analysing these evidentiary transitions and the assumptions embedded within them. Though the *Presentation* and *Delivery* models primarily concern the logistical and environmental implementation of assessment, the remaining core structural models provide a basis for evaluating the quality and transparency with which assessment approaches are described in the literature.

3 Methodology

Our work employed a mixed-methods approach that combined a Systematic Literature Review (SLR) with long-form semi-structured interviews. This combination allowed us to integrate insights from existing research on assessment approaches for group projects in computing with perspectives from current practitioners.

The SLR was conducted following the guidelines proposed by Kitchenham [2004], complemented by recommendations from more recent advice including Petersen et al. [2015] and the PRISMA 2020 statement [Page et al. 2021]. Its primary aim was to map the range of assessment approaches for group projects in computing, with an emphasis on studies relevant to post-secondary education. To achieve this, we developed the review protocol, executed systematic searches, screened and assessed the quality of the studies, and extracted and synthesised data to answer the research questions.

In parallel, we conducted semi-structured interviews with computing instructors to capture perspectives and practices that might not be fully represented in the published literature. The interview process consisted of constructing the interview script and creating the pre-interview survey, identifying participants, conducting the interviews, performing thematic analysis following Braun and Clarke's [2006] six-stage thematic analysis framework, extracting relevant quotes and distilling recommendation, and synthesising the findings and connecting these back to the results from the literature.

When reporting our findings, we bring together the results from the SLR and interviews in an integrated manner. The results from the SLR informed the development of the interview protocol and enabled us to use prior research to explain participants' responses. Meanwhile, the interview insights clarified and extended results observed in the literature. Taken together, these methods provide a more comprehensive and practical information on assessment approaches for group projects in computing.

3.1 Systematic Literature Review

3.1.1 Search and Screening Process. The review protocol was developed collectively by the research team and subsequently reviewed by the Research Ethics and Integrity Committee at the first author's institution. The protocol incorporated feedback from this review. As part of this process, the protocol was verified to accord with the ethical principles of the British Educational Research Association (BERA) [Wyse et al. 2016]. The first author secured ethical approval through the formal ethics process at their institution (as did other authors where required).

The date range, beginning in 2013, coincides with the publication of the ACM/IEEE Task Force CS2013 recommended curricula[2013], which significantly emphasised collaborative and project-based learning in computing education. This period also corresponds with the adoption of platforms like Slack and GitHub [Zagalsky et al. 2015], now commonplace in computing group projects.

To support collaborative coding, bibliographic records were maintained in `.bib` format using JabRef 5.13, version-controlled via a private Git repository. Extracted attributes and codes were tabulated and managed in Microsoft Excel, in Office 365, enabling the team to conduct structured comparison across studies.

The authors searched five digital libraries: ACM Digital Library, IEEE Xplore, Sage Journals, ScienceDirect, and Taylor & Francis Online. The search query was:

> *[[All: "group project"] OR [All: "group assessment"] OR [All: "group assignment"] OR [All: "team project"] OR [All: "team assessment"] OR [All: "team assignment"] OR [All: "peer assessment"] OR [All: "team work"]] AND [[All: "computer science education"] OR [All: "computing education"] OR [All: "software engineering education"] OR [All: "csed"] OR [All: "cs ed"]].*

In some databases, the search additionally included abstracts, as these databases index search queries across titles, keywords, and abstracts by default. An initial trial application of the search string to a test set of articles resulted in minor refinements, notably the explicit inclusion of "peer assessment" due to its omission in early results.

The following inclusion criteria were applied:

- **IC1**: Original papers published between 2013 and 2024
- **IC2**: Papers published in peer-reviewed conference proceedings and journals
- **IC3**: Papers published in English
- **IC4**: Papers where the body was at least four pages in length
- **IC5**: Papers presenting empirical evidence through a primary study
- **IC6**: Papers describing the use of group projects at a post-secondary institution in a computing discipline
- **IC7**: Papers which include a description of an approach to assessing a group project

Gray literature, such as dissertations, technical reports, and non-peer-reviewed sources, was excluded. These sources frequently lacked sufficient methodological transparency and detail about assessment approaches, limiting their utility for structured synthesis in a systematic review of this kind.

The initial search query produced 1221 papers. Nine authors independently screened titles and abstracts against the inclusion criteria for these papers. Full-text screening and data extraction were similarly distributed. To ensure consistency in the application of inclusion criteria and data extraction categories, a Krippendorff's alpha reliability coefficient was computed on a random 10%

sample of papers reviewed by multiple authors, achieving a coefficient of 0.8, indicating substantial agreement [Krippendorff 2018].

Quality appraisal was conducted using two complementary strategies. First, each paper was assessed for empirical robustness using criteria from the 2018 Mixed Methods Appraisal Tool (MMAT) [Hong et al. 2018]. It is a critical appraisal instrument designed for the systematic assessment of the methodological quality of empirical studies encompassing qualitative, quantitative, and mixed methods research designs [Hong et al. 2018]. There are five distinct sets of criteria corresponding to different study types: qualitative, randomised controlled trials, non-randomised studies, quantitative descriptive studies, and mixed methods studies. The tool facilitates a structured and transparent appraisal process by guiding reviewers in assessing key aspects such as appropriateness of study design, sampling strategy, measurement tools, and data integration. Second, the descriptions of assessment approaches within each study were appraised using Almond's assessment framework [Almond et al. 2002] that identifies four essential processes in any assessment system: activity selection, presentation, response processing, and summary scoring. These processes collectively form the assessment cycle, which involves two key actors: the administrator, responsible for setting up and managing the assessment, and the participant, whose skills are being assessed. The participant interacts with tasks presented during the assessment. Moreover, Almond's framework highlights six integral components of assessment: student, evidence, task, assembly, presentation, and delivery. These components correspond to the desired learning outcomes, the interpretation of evidence to measure those outcomes, the tasks that generate this evidence, the instructions linking student, task, and evidence, the medium or format for presenting the assessment, and the overarching framework that integrates these elements in practice. We used four of the six components, *Student*, *Task*, *Evidence*, and *Assembly*, in Almond's framework to categorise our findings because they explore the core elements that impact the design and interpretation of group assessments. By concentrating on these four components, we can examine the foundational aspects of assessment approaches that can be generalised and adapted across various educational contexts.

3.1.2 Data Extraction and Synthesis. Data extraction followed a staged content analysis approach [Krippendorff 2018], combining both inductive and deductive elements. Initially, a subset of papers (n=12) was reviewed independently by members of the research team to identify recurring themes, terms, and concepts related to the assessment of student group projects. Through open coding [Cascio et al. 2019] and collaborative discussion, a preliminary codebook was developed to capture dimensions such as assessment focus, modality, and rationale. This codebook was iteratively refined over multiple rounds of review, during which codes were merged, split, or redefined to reflect better the patterns observed across the literature. Once the codebook was stabilised, it was applied deductively to the remaining corpus.

Data were extracted under consistent categories, the columns of the spreadsheet, including:

- the setting of the group project and its position within the curriculum,
- national contexts (using the United Nations Statistics Division regional mapping[1]),
- computing subdiscipline or knowledge area,
- stated learning outcomes and assessment rationale,
- articulation of the assessment approach,
- assessment characteristics [Vasilevskaya et al. 2014], such as formative vs. summative, and group-grade vs. individual-grade structures,
- additional implementation details, including team sizes, peer evaluation mechanisms, and technology use,
- tools and how they were used in the study, and

[1] https://unstats.un.org/sdgs/indicators/regional-groups/

- evidence supporting the approach's effectiveness.

In addition, the intended learning outcomes were extracted using the model proposed by Herbert [2018]. This illustrated whether the assessment focused upon the product, the process, the professionalism, or the progress that students made with their learning. There were some instances where fundamental computing knowledge was assessable, and so an additional categorization was added: Principles.

The schema of 23 assessment modes emerged through an inductive synthesis process based on the coded data. Following the standards of qualitative content analysis [Mayring 2014], the research team repeatedly reviewed coded entries related to assessment methods to identify patterns, similarities, and differences in approach. This involved grouping similar practices, distinguishing edge cases, and refining category labels to ensure mutual exclusivity and conceptual clarity.

The resulting schema was not predetermined but developed iteratively through constant comparison and collaborative discussion, informed by both the extracted data and the researchers' shared understanding of pedagogical assessment in computing. While not derived from grounded theory per se, the process aligns with an interpretive, constructivist orientation, in which categories were generated from the data rather than imposed a priori. The final schema, encompassing twenty-three modes of assessment, serves as an organising framework for reporting findings in this review.

3.2 Interviews

3.2.1 Instruments. The interview protocol of 22 open-ended questions was created with some high-level input from the SLR team by a small subteam of researchers experienced in scripting and conducting semi-structured interviews and constructing survey studies, paying careful attention to the neutral phrasing of prompts to avoid leading the participants and biasing responses. The pre-interview survey collected background and demographics on interviewees and asked for their consent to participate in a recorded interview. The interview questions included prompts about group project assessment strategies specific to the interviewee, about personnel, individual vs collective assessment, team dynamics, tools, fairness, student feedback, peer evaluation, institutional context, challenges, future directions, and asked the participants to give advice to other instructors about things to do and things not to do when assessing group work in computing courses. Both instruments were reviewed and received ethical approval by the Institutional Review Boards of each of the four interviewers belonging to institutions in the USA, UK, Sweden and Germany.

3.2.2 Participants. Members of the research team following a network-based sampling strategy identified potential candidates for interview and invited them to participate via email. Upon acceptance, they were then introduced to another member of the team to conduct the interview, as a measure to minimise acquaintance bias.

Each interviewee had experience with undergraduate courses that involved students working as a team to deliver a project. Only those who had at least one full cycle/delivery of a team project-based course were eligible. On that basis, the research team interviewed 18 instructors at higher education institutions across ten countries. These included Australia (2), Canada (3), China (1), England (2), Germany (2), The Netherlands (1), New Zealand (2), Scotland (1), Sweden (1), and the USA (3). One of the interviewees had 1-2 years of experience conducting group project courses, four had between 3 and 5 of experience, and the remaining thirteen participants have been involved with group project courses for 5 or more years. Twelve of the interviewees were male, five were female and one was gender non-binary.

Interviews were done remotely in English using Microsoft Teams or Zoom. Sessions were recorded and transcribed by the built-in meeting software transcription function and manually

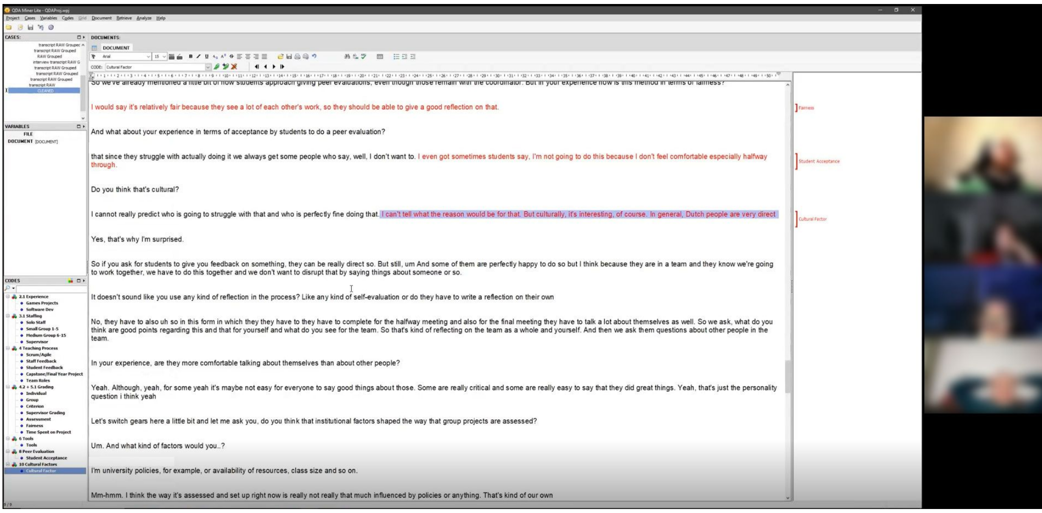


Fig. 3. Using QDAMiner to Identify Codes

checked for accuracy and corrected where necessary. A little over 22 hours of video footage was gathered and analysed.

3.2.3 Qualitative Data Analysis and Synthesis. The transcripts were analysed following a minor adaptation to Braun and Clarke's [2006] six-stage thematic analysis framework. This process seeks to identify and articulate the thematic elements which arose—defined as patterns of meaning that reflect common concepts across instances or examples. The analysis commenced with data familiarisation and identifying the overarching topics from the data collected during interviews. Four members of the research team agreed on the most common themes across the overall corpus. Then the transcripts were inductively coded [Peel 2020] using QDA Miner[2] as shown in Figure 3. Following the construction of the codebook, each transcript was then coded in depth by two researchers selected at random. These raters identified additional potential themes independently as necessary.

Following this initial thematic review, the team convened, including all authors, to discuss their findings and work towards reconciling their codes into a coherent codebook. After reaching a consensus, two team members independently re-coded the data according to this unified framework.

The final dataset underwent a thorough review to highlight any lingering points of contention among the coders, with further consultations involving the remaining authors to resolve any stalemates. This consensus coding process was iterated until it achieved at least 80% agreement on 95% of the codes [Miles et al. 2014]. The approach deepened the group's understanding of each theme's definition, scope, and distinctions in complex cases. This approach enabled a thorough analysis beyond a single pass and minimised clerical errors in coding.

In parallel, the team identified note-worthy quotes to be included in the manuscript and distilled recommendations for other instructors given by the interviewees. Once completed, the final analysis allowed for the examination of the relative prevalence of each theme within the dataset while considering inter-rater concordance using Krippendorff's alpha [Hayes and Krippendorff 2007].

[2]https://provalisresearch.com/products/qualitative-data-analysis-software/freeware/

3.2.4 Additional Data Extraction and Reconciliation. After the qualitative analysis was complete, three of the researchers involved in the thematic analysis independently re-watched the interview footage and recorded the time codes and content meeting learning outcome definitions as outlined in Table 1. Our analysis aligns with how we evaluated the SLR papers for the various dimensions of assessment: principles; progress; product; process and professionalism. The team then met to discuss and reconcile their findings and agreed on which assessment activities were mentioned as utilised by each of the interviewees.

4 Systematic Literature Review Results

In this section, we present our SLR results, starting with Section 4.1, presenting an overview of the papers identified by our selection criteria. Section 4.2 provides details on the assessment approaches found in these papers.

4.1 Overview of Selected SLR Papers

In this section, we report on the results from our Systematic Literature Review (SLR). Of the original 1,221 papers published from 2013 to 2024, 44 (3.7%) met our selection and quality criteria, which we aggregated by publication year, shown in Figure 5. As shown in Figure 4, we removed 1177 papers from our SLR because:

- Four were duplicates,
- Five were not peer-reviewed,
- 21 were shorter than four pages,
- 365 did not contain a primary study, such as a literature review or position paper,
- 621 did not focus on group project assessment,
- 145 conducted the studies outside of a tertiary institution,
- Seven did not provide sufficient information on the assessment approach, and
- Nine did not provide enough information to determine how the work aligns with Almond's Framework.

When reviewing the selected literature, we examined the countries in which the research was conducted, their context, and the group size. For location, we found that most ($N = 28$, 64%) were conducted in Europe and North America. The remaining papers come from Australia and New Zealand ($N = 6$, 14%), Eastern and South-Eastern Asia ($N = 3$, 7%), Latin America and the Caribbean ($N = 5$, 11%), Northern Africa and Western Asia ($N = 1$, 2%), and Central and Southern Asia ($N = 1$, 2%). Unfortunately, our SLR did not find studies conducted in Sub-Saharan Africa. For the context, most ($N = 27$, 61%) of the papers in the SLR conducted the research in a Software Engineering context, followed by ($N = 7$, 16%) placed in a Capstone context. The next common context ($N = 4$, 9%) was Employability and Professionalism, referring to courses focused on preparing students for the workplace rather than on technical computing content, followed by the Specialist Topic context ($N = 4$, 9%), which included courses focused on a specific advanced computing subject area, such as natural language processing. The remaining two papers (5%) conducted studies in the Computing Education context, focusing on the application of rubrics in Computing Education [Kavanagh and Luxton-Reilly 2016] and the perception of peer evaluation and fairness [Aivaloglou and Meulen 2021]. The majority ($N = 37$, 84%) of the papers reported group size, with a median of five students, a minimum of three students in three papers, and a maximum of 35 students from a study by Vasilevskaya et al. [2015] that applied their work to a course that leverages large projects for student learning. The study collected feedback over two years on the assessment using a questionnaire. The researchers presented their findings on how to achieve fair grading in large student groups, primarily using summative activities, such as individual student reports and teacher assessments.

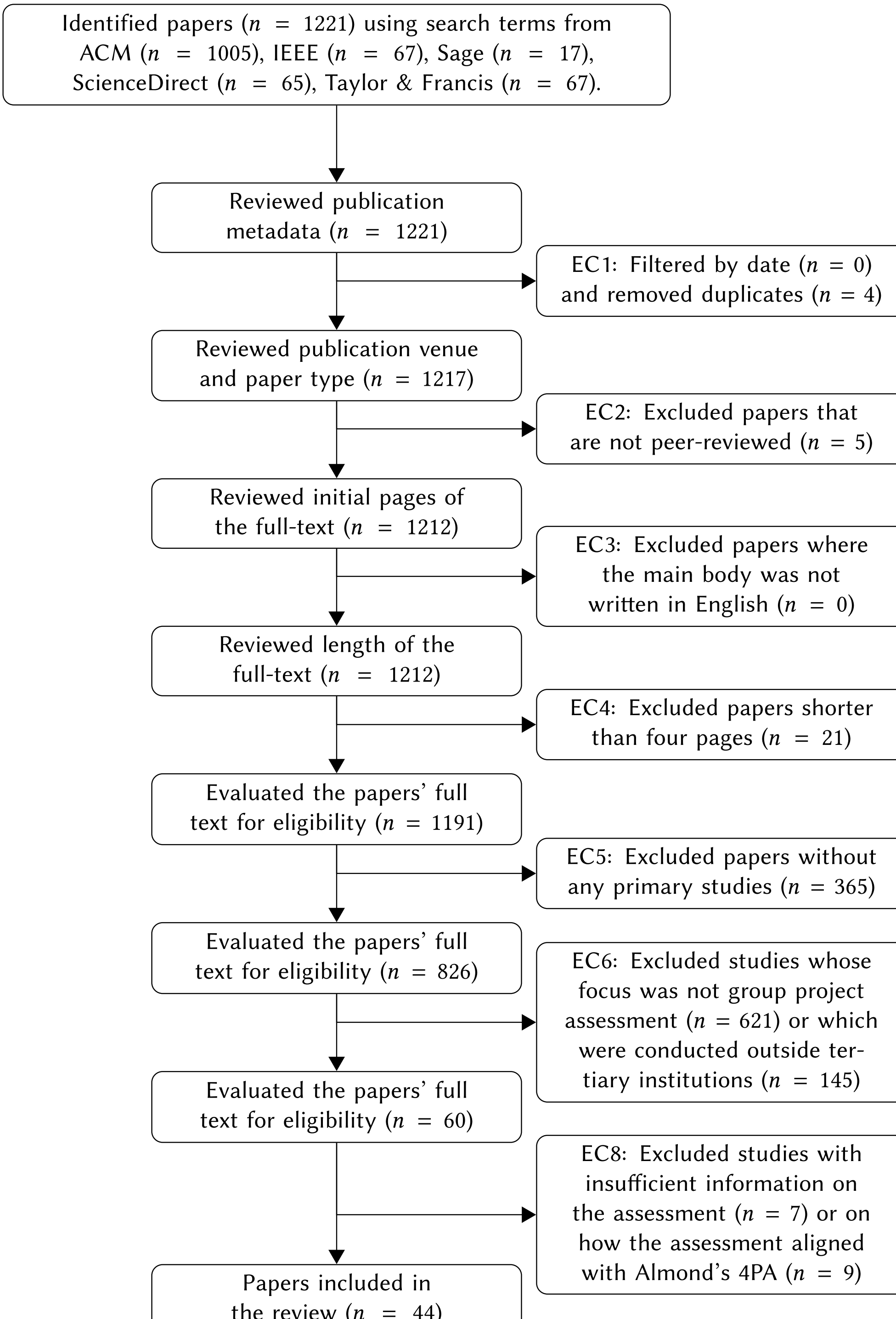


Fig. 4. Process of selecting papers for the content analysis

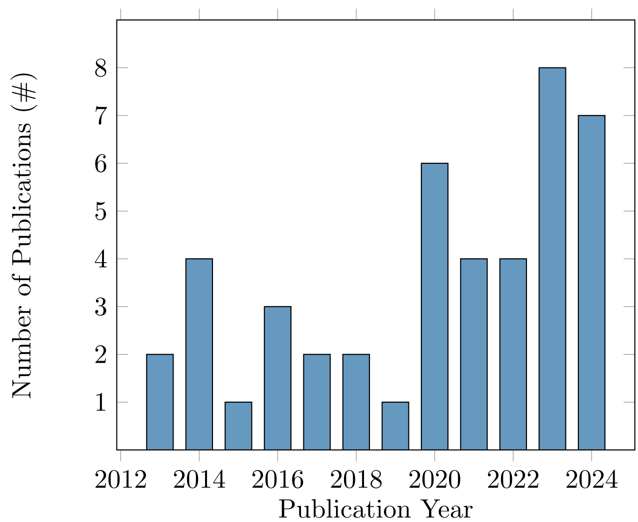


Fig. 5. Count of publications describing the assessment of student teamwork in computing per year

We evaluated the papers in more depth to examine their assessment types and focus. We found that the majority ($N = 43$, 98%) of the papers presented the assessment types used in the studies. For the grading approach, 27 (61%) used summative assessment, six (14%) used formative, and ten (23%) used both assessment types. As for the assessment focus, 26 (59%) of the studies assessed the individual, four (9%) focused on group, and 10 (23%) assessed both.

We also examined the papers to determine the tools used to support the assessment approach. We found 24 (56%) papers that used tools. Table 4 provides a breakdown of the purpose of the tools used in the assessment approach, along with examples of the tools. Although the majority ($N = 22$, 50%) of the papers did not use tools, the most common usage ($N = 11$, 25%) was for collaboration, utilising tools such as GitHub. One paper [Hundhausen et al. 2023] appears in multiple rows of Table 4 as it used tools for three purposes: to support student collaboration, to facilitate peer assessment, and to support educators in managing group work. In these papers, educators utilised various tools, including SPARK [Kay 2022], Microsoft and Google Forms [Hijazi and Alshehri 2024; Wee Tan et al. 2023], and PASCA [Neznanov and Maksimenkova 2016], for peer assessment. PASCA is a mail-based tool, and SPARK (Summative Peer Assessment while Rounding Kindly) both facilitate anonymous peer assessment to encourage fair evaluation by the students. We also observed studies using Software Engineering tools, such as GitHub [Torchiano et al. 2024], Docker Hub [Torchiano et al. 2024], Slack and Microsoft Teams [Hundhausen et al. 2023], to support team coordination. Other tools, such as ePortfolios [Olstad et al. 2024], supported students' self-reflection during the teamwork activity.

As previously mentioned in our study design, Section 3, we evaluated the assessment approaches in the selected literature against four components of Almond's framework [Almond et al. 2002]. Table 5 shows how the papers in the SLR aligned with Almond's framework. The table shows how the papers address each of Almond's four selected components: *Student*, *Evidence*, *Task*, and *Assembly*. Of the 53 papers that passed the initial selection criteria, we excluded nine following quality appraisal due to insufficient detail about the assessment approach employed, leaving 44

Table 4. View of the Literature Examining Tools use to Support the Assessment Approach

#	Purpose*	Example	References
22	No Tool Mentioned		[Avila et al. 2021; Bhat et al. 2020; Choque-Soto and Sosa-Jauregui 2024; Christov and Hoffman 2019; Fagerholm and Vihavainen 2013; Giacaman and Sinnen 2018; Guttmann et al. 2024; Isomöttönen 2014; Isomöttönen and Ritvos 2021; Kaluza et al. 2023; Kavanagh and Luxton-Reilly 2016; Mora et al. 2020; Neill et al. 2017; Porquet-Lupine and Brigham 2023; Putro et al. 2014; Quek et al. 2023; Tsai et al. 2023; Tubino et al. 2020; Vasilevskaya et al. 2014, 2015; Wee Tan et al. 2023; Yun et al. 2016]
11	Collaboration	Google Community Tools, GitHub	[Aivaloglou and Meulen 2021; Buffardi 2020; Cecilia Bastarrica et al. 2023; Chatley and Field 2017; Cui et al. 2022; Herbert 2018; Hijazi and Alshehri 2024; Hundhausen et al. 2023; Jacobs 2024; Neyem et al. 2014; Torchiano et al. 2024]
8	Peer Assessment	PASCA: Mail Peer Assessment System	[Basholli et al. 2013; Candrlic 2020; Crick et al. 2022; Hundhausen et al. 2023; Kay 2022; Lu and van der Hoek 2023; Neznanov and Maksimenkova 2016; Vossen and Ajit 2020]
3	Educator Support	CATME Team Builder	[Chen et al. 2022; Hundhausen et al. 2023; Tubino et al. 2021]
1	Real-World Experience	JIRA	[Allison et al. 2024]
1	Self Reflection	ePortfolio	[Olstad et al. 2024]

*Purpose denotes how a tool was used in the study

(83%) for further analysis. Of these 44 papers, 15 (28%) provided complete information across all components of Almond's framework. Among the four Almond components, *Evidence* had the most papers ($N = 39$, 74%) that addressed it.

4.2 Assessment Approaches

Lastly, seeking to answer "**RQ1**: *What assessment approaches are typical of student group projects in undergraduate computing education?*" we examined the SLR papers by their learning outcomes and categorised them by assessment dimension: *Product*, *Process*, *Professionalism*, *Progress* and our addition, *Principals*. Table 6 presents the papers in each category along with the assessment activity used in the studies to evaluate learning outcomes. While examining the papers, we found a confluence between the *Process* and *Professionalism* categories, where 35 papers focused on both learning outcomes. For example, Allison et al. [2024] addresses *Process* and *Professionalism* by incorporating Agile methodologies, specifically the SCRUM framework, into the curriculum. In this study, the researchers simulated real-world industry practices, requiring students to work collaboratively. The study's context mirrors professional environments, helping students develop

Table 5. View of the Literature Through the Almond's Assessment Framework

Almond's Components				**Papers**	
Student	**Evidence**	**Task**	**Assembly**	**#**	**References**
✓	✓	✓	✓	15	[Aivaloglou and Meulen 2021; Allison et al. 2024; Avila et al. 2021; Bhat et al. 2020; Chen et al. 2022; Fagerholm and Vihavainen 2013; Herbert 2018; Hundhausen et al. 2023; Isomöttönen and Ritvos 2021; Mora et al. 2020; Neznanov and Maksimenkova 2016; Porquet-Lupine and Brigham 2023; Putro et al. 2014; Tsai et al. 2023; Vasilevskaya et al. 2015]
✓	✓	✓		1	[Choque-Soto and Sosa-Jauregui 2024]
✓	✓	✓		3	[Candrlic 2020; Quek et al. 2023; Torchiano et al. 2024]
✓	✓		✓	2	[Cecilia Bastarrica et al. 2023; Giacaman and Sinnen 2018]
✓		✓	✓	2	[Basholli et al. 2013; Isomöttönen 2014]
	✓	✓	✓	2	[Tubino et al. 2020, 2021]
✓	✓			3	[Jacobs 2024; Neill et al. 2017; Olstad et al. 2024]
✓		✓		1	[Kaluza et al. 2023]
	✓	✓		4	[Buffardi 2020; Christov and Hoffman 2019; Kay 2022; Neyem et al. 2014]
✓				1	[Yun et al. 2016]
	✓			9	[Crick et al. 2022; Cui et al. 2022; Guttmann et al. 2024; Hijazi and Alshehri 2024; Kavanagh and Luxton-Reilly 2016; Lu and van der Hoek 2023; Vasilevskaya et al. 2014; Vossen and Ajit 2020; Wee Tan et al. 2023]
		✓		1	[Chatley and Field 2017]
28	39	29	21	44	**Total**†

† Total number of papers addressing a component in Almond's framework

process management skills and professional behaviours, such as teamwork, communication, and adaptability. To show papers addressing *Process* and *Professionalism*, we present them together in Table 6, where the majority ($N = 35$, 80%) of the papers focus on these learning outcomes. The table also shows that *Principals* have the least papers ($N = 6$, 14%) focusing on this learning outcome.

The summary of the SLR shown in Table 6 illustrates that the focus of assessment varied in terms of learning outcome. For example, the previously mentioned study by Allison et al. [2024] addressing *Process* and *Professionalism* also addresses the *Product* learning outcome. They incorporate Agile methodologies to also enhance students' employability skills by having the group-based projects deliver a product that meets the client's requirements. This hands-on experience supports students in applying theoretical software engineering concepts associated with Agile to create tangible, functional products.

To evaluate these learning outcomes, the papers use a variety of approaches, as shown in the first column of Table 6. The table shows commonly used assessments applied in learning environments, such as exams, exercises, and reports, to evaluate learning outcomes. These approaches evaluate the individual (e.g., exams) as well as evaluate the group (e.g., presentations). In addition to assessment approaches commonly used in learning environments, the results show papers using real-world activities, such as peer review, build inspection, and postmortems, to assess learning outcomes. A postmortem, sometimes referred to as *retrospective*, is a reflective analysis conducted after the completion of a project or an event [Kerzner 2017]. Overall, the results show papers using peer evaluation for assessing learning outcomes the most ($N = 19$, 43%).

Table 6. View of the learning outcomes in the literature through the 5 P's

Approach	# of Papers	Reference
Principles: ($n = 6$, 14%)		
Exam	4	[Giacaman and Sinnen 2018; Kaluza et al. 2023; Neill et al. 2017; Porquet-Lupine and Brigham 2023]
Home Exercise	2	[Neill et al. 2017; Yun et al. 2016]
Presentation	1	[Bhat et al. 2020]
Progress: ($n = 11$, 25%)		
Reflective Review	8	[Allison et al. 2024; Basholli et al. 2013; Choque-Soto and Sosa-Jauregui 2024; Christov and Hoffman 2019; Herbert 2018; Isomöttönen 2014; Isomöttönen and Ritvos 2021; Jacobs 2024]
Retrospective	1	[Torchiano et al. 2024]
Exam + Portfolio	1	[Olstad et al. 2024]
Portfolio	1	[Quek et al. 2023]
Product: ($n = 23$, 52%)		
Presentation	11	[Basholli et al. 2013; Candrlic 2020; Cecilia Bastarrica et al. 2023; Giacaman and Sinnen 2018; Herbert 2018; Isomöttönen and Ritvos 2021; Kaluza et al. 2023; Neyem et al. 2014; Torchiano et al. 2024; Tsai et al. 2023; Yun et al. 2016]
Build Inspection	8	[Giacaman and Sinnen 2018; Herbert 2018; Hijazi and Alshehri 2024; Neill et al. 2017; Neyem et al. 2014; Porquet-Lupine and Brigham 2023; Torchiano et al. 2024; Tsai et al. 2023]
Report	3	[Christov and Hoffman 2019; Jacobs 2024; Vasilevskaya et al. 2014]
Peer Review	2	[Mora et al. 2020; Neznanov and Maksimenkova 2016]
Portfolio	2	[Tubino et al. 2020, 2021]
Documentation	3	[Allison et al. 2024; Basholli et al. 2013; Neyem et al. 2014]
Duels	1	[Yun et al. 2016]
Structured Exercises	1	[Chatley and Field 2017]
Process + Professionalism: ($n = 35$, 80%)		

Continued on next page...

Table 6 – Continued from previous page

Approach	# of Papers	Reference
Peer Evaluation	20	[Aivaloglou and Meulen 2021; Avila et al. 2021; Basholli et al. 2013; Buffardi 2020; Candrlic 2020; Cecilia Bastarrica et al. 2023; Crick et al. 2022; Fagerholm and Vihavainen 2013; Herbert 2018; Hijazi and Alshehri 2024; Hundhausen et al. 2023; Jacobs 2024; Kavanagh and Luxton-Reilly 2016; Kay 2022; Lu and van der Hoek 2023; Neyem et al. 2014; Porquet-Lupine and Brigham 2023; Vasilevskaya et al. 2015; Vossen and Ajit 2020; Wee Tan et al. 2023]
Analytics	9	[Aivaloglou and Meulen 2021; Buffardi 2020; Chen et al. 2022; Choque-Soto and Sosa-Jauregui 2024; Cui et al. 2022; Guttmann et al. 2024; Hundhausen et al. 2023; Putro et al. 2014; Torchiano et al. 2024]
Structured Observation	5	[Basholli et al. 2013; Cecilia Bastarrica et al. 2023; Herbert 2018; Hundhausen et al. 2023; Kaluza et al. 2023; Neyem et al. 2014; Vasilevskaya et al. 2014]
Scaffolded Discussion	2	[Isomöttönen 2014; Neill et al. 2017]
Panel	2	[Tubino et al. 2020, 2021]
Portfolio	2	[Allison et al. 2024; Isomöttönen and Ritvos 2021]
Interview	2	[Porquet-Lupine and Brigham 2023; Vasilevskaya et al. 2014]
Postmortem	1	[Christov and Hoffman 2019]

For the remainder of this section, our modified schema of four of the five P's, *Principals*, *Progress*, *Product*, and *Process + Professionalism*, will be discussed in greater detail. The following subsections outline the assessment approaches we observed and present the different modalities inferred in these papers, along with the modalities associated with different dimensions of assessment.

4.2.1 Principals. This domain measures the student's grasp of fundamental computing concepts, as well as their understanding of context-specific knowledge relevant to the project at hand. It assesses not only what students know but also how they can apply that knowledge effectively in practical scenarios.

Exam

Exams are often used as a controlled method to assess individual student knowledge in a time-constrained environment. In computing education, exams may involve problem-solving, programming tasks, or theoretical questions about core computing concepts, such as algorithms, data structures, or software engineering principles. The exam can serve as an effective tool for evaluating how well students grasp the theoretical aspects of computing knowledge in a team-based setting, such as understanding how to approach collaborative programming or distributed system design under pressure.

Exams may also be designed to assess how students engage with complex problems individually before collaborating with teammates, ensuring that they possess the foundational knowledge necessary to contribute meaningfully to a team project [Giacaman and Sinnen 2018; Kaluza et al. 2023; Neill et al. 2017; Porquet-Lupine and Brigham 2023].

Home Exercise

Lab assignments and homework are practical assessments that allow students to demonstrate their skills in solving problems or building solutions. These assignments often come in the form of small programming tasks, quizzes, or practical exercises that help students apply the concepts they have learned in class. In the context of teamwork, these tasks are valuable for assessing both individual and group-based learning outcomes. Lab or homework assignments can test the team's collective ability to approach and resolve technical challenges, fostering collaboration through pair or group work. In addition, they provide a structured way to track progress and pinpoint areas where students may need further support [Neill et al. 2017; Yun et al. 2016].

Presentation

Presentations offer a dynamic method for students to articulate their knowledge and understanding of a particular topic. In computing education, presentations often involve explaining a technical concept, demonstrating a solution to a problem, or presenting the outcomes of a project. As part of teamwork assessments, presentations are used to evaluate how well students can communicate their ideas, defend their reasoning, and explain the impact of their project to an audience. This is particularly important for assessing teamwork skills, as communication and presentation are essential in collaborative settings where individuals must convey their perspectives to achieve consensus or share progress with stakeholders [Bhat et al. 2020].

4.2.2 Progress. This domain focuses on the introspective aspect of the learning experience, typically a reflection. The domain encourages students to critically evaluate both their individual and collective performance, identifying their strengths and weaknesses. This domain helps students pinpoint areas where they can improve and develop in the future, fostering a mindset geared towards continuous learning and growth.

Reflective Review

A reflective review involves writing about one's experiences during the project, focusing on what they learned and how they overcame challenges. This type of assessment encourages students to think critically about their personal growth and development during the project. Reflective reviews help assess how well students understand their own skills, acknowledge their strengths and weaknesses, and plan for future improvement [Allison et al. 2024; Basholli et al. 2013; Choque-Soto and Sosa-Jauregui 2024; Christov and Hoffman 2019; Herbert 2018; Isomöttönen 2014; Isomöttönen and Ritvos 2021; Jacobs 2024].

Retrospective

A formal recording of a sprint retrospective, or other similar review of work-in-progress and/or the team dynamic at the end of a period of development [Torchiano et al. 2024].

Exam + Portfolio

In this context, an exam might be structured as a reflective review but conducted under formal exam conditions. This type of assessment can focus on individual reflections about their experiences in the team, their contributions to the project, and the lessons learned throughout the process. It provides an opportunity for students to consolidate their learning in a structured way under controlled circumstances [Olstad et al. 2024].

Portfolio

A team portfolio captures a longitudinal record of team health by having students complete repeated rubric-based assessments throughout the project, tracking dimensions such as team dynamics, communication, and collaboration. This ongoing collection of evidence helps identify where the team performed well and where it needs improvement, offering students a structured basis to reflect on teamwork effectiveness and guide their development across successive iterations [Quek et al. 2023].

4.2.3 Product. This domain is centred on the final output of the group, whether that be a goal milestone, completed project, a tangible artefact, or the presentation of a solution. The primary goal here is to assess the quality and effectiveness of the results produced by the team, determining how well they have met the project's objectives and standards.

Presentation

A presentation or demonstration of the project's outcome, whether oral or video-based, is another way to assess the final deliverable. Presentations enable students to showcase their work, explaining the technical aspects, challenges encountered, and how they overcame them. This method evaluates not only the technical quality of the solution but also the student's ability to communicate the solution clearly and effectively to an audience [Basholli et al. 2013; Candrlic 2020; Cecilia Bastarrica et al. 2023; Giacaman and Sinnen 2018; Herbert 2018; Isomöttönen and Ritvos 2021; Kaluza et al. 2023; Neyem et al. 2014; Torchiano et al. 2024; Tsai et al. 2023; Yun et al. 2016].

Build Inspection

Build inspection refers to assessing the compiled build, release, or source code produced by the team. This assessment method can involve automated testing (e.g., unit tests) or manual inspection of the code, such as checking for errors, inefficiencies, or adherence to coding standards. It helps evaluate the technical quality of the solution and ensures that the final product is functional, reliable, and meets the specified requirements [Giacaman and Sinnen 2018; Herbert 2018; Hijazi and Alshehri 2024; Neill et al. 2017; Neyem et al. 2014; Porquet-Lupine and Brigham 2023; Torchiano et al. 2024; Tsai et al. 2023].

Report

A report is a written document detailing the student's acquired knowledge on an assigned topic. These reports are typically administered at the end of a project or semester to gather feedback from students regarding their experiences working in teams. It often includes questions about how well the team worked together, the challenges faced, and individual reflections on the collaborative process. This method can help assess students' perceptions of the project, the teamwork process, and their personal growth throughout the course. Reports may also provide valuable data for instructors to adjust their teaching methods or project designs in future courses [Christov and Hoffman 2019; Jacobs 2024; Vasilevskaya et al. 2014].

Peer Review

Peer review involves students reviewing the work of their peers, such as team members in a learning activity. Often, this was code, and would constitute a *merge request.* This method helps assess the quality of the deliverables from multiple perspectives and promotes critical thinking. Peer review encourages students to evaluate the work of their teammates, providing constructive feedback that can enhance the final product. It also fosters a deeper understanding of the work being done, as students must engage with the content at a detailed level to evaluate its strengths and weaknesses [Mora et al. 2020; Neznanov and Maksimenkova 2016].

Portfolio

In the context of our study, a portfolio refers to a compilation of various deliverables produced throughout the project. It may include source code, progress logs, design documents, user manuals, and other relevant materials that reflect the team's work. A portfolio provides a comprehensive view of the group's achievements and helps instructors assess the completeness, quality, and diversity of the outputs produced during the project [Tubino et al. 2020, 2021].

Documentation

Documentation is a critical component in evaluating the quality and usability of the final project. This can include user manuals, API documentation, or technical specifications that explain how the project works and how others can use or maintain it. Well-documented projects are easier to maintain, debug, and extend, making documentation a key factor in evaluating the overall quality of the deliverable [Allison et al. 2024; Basholli et al. 2013; Neyem et al. 2014].

Duels

A team debate, which involves discussing the project's outcomes and evaluating the different approaches or solutions considered during its development. This type of assessment fosters critical thinking and collaborative discussion, as team members must justify their decisions and reflect on the effectiveness of their strategies. Debates are also useful for assessing how well students understand the implications of their decisions and the trade-offs involved in developing software solutions [Yun et al. 2016].

Structured Exercises

Structured exercises involve completing specific tasks according to predefined specifications. In computing education, these exercises often test the performance of a solution or application rather than the solution itself. For example, exercises might include evaluating how well a program performs under specific conditions or how efficiently a system meets user requirements. This form of assessment is valuable for testing the real-world applicability of the team's work and ensuring that the solution meets the required performance standards [Chatley and Field 2017].

4.2.4 Process and Professionalism. In this domain, the focus is on evaluating the professionalism and teamwork abilities of students. It encompasses critical skills such as effective project management, communication among team members, and the overall synergy of the group as they strive to meet project objectives. Assessments in this domain also aim to provide insight into how individual contributions are recognised within the context of a collaborative effort, ensuring that every team member's role is accounted for.

Peer Evaluation

Peer evaluation involves students providing feedback or ratings on the performance of their peers. This assessment approach allows students to reflect on the contributions of their teammates, offering insights into how effectively they collaborated and whether all team members met expectations. Peer evaluations often include self-assessments and statistics on relative contributions, which can help identify the strengths and weaknesses of individual team members. This form of assessment also promotes accountability, as students are more likely to actively participate in team projects when they know their peers will evaluate them [Aivaloglou and Meulen 2021; Avila et al. 2021; Basholli et al. 2013; Buffardi 2020; Candrlic 2020; Cecilia Bastarrica et al. 2023; Crick et al. 2022; Fagerholm and Vihavainen 2013; Herbert 2018; Hijazi and Alshehri 2024; Hundhausen et al. 2023; Jacobs 2024; Kavanagh and Luxton-Reilly 2016; Kay 2022; Lu and van der Hoek 2023; Neyem et al. 2014; Porquet-Lupine and Brigham 2023; Vasilevskaya et al. 2015; Vossen and Ajit 2020; Wee Tan et al. 2023].

Analytics

Analytics involves using tools such as commit logs, burndown charts, project management systems (e.g., task boards), and ticketing systems to track the progress and individual contributions of team members in a project. By analysing such data, instructors can gain insights into how actively each student is involved in the project, their coding contributions, and the quality of their collaboration with other team members. This method can help attribute specific tasks to individual team members, making it easier to assess both the quality and quantity of their contributions. It also allows for real-time monitoring of project progress and the identification of potential issues in team dynamics [Aivaloglou and Meulen 2021; Buffardi 2020; Chen et al. 2022; Choque-Soto and Sosa-Jauregui 2024; Cui et al. 2022; Guttmann et al. 2024; Hundhausen et al. 2023; Putro et al. 2014; Torchiano et al. 2024].

Structured Observation

Structured observation involves an instructor or observer making detailed notes about the team working in a specific setting or context, often during team meetings. Observations may focus on interactions between team members, how they resolve conflicts, the process by which decisions are made, and whether the group adheres to a defined workflow. This method offers valuable insight into the team's communication dynamics, problem-solving strategies, and the individual contributions of each team member. It also provides opportunities to assess soft skills such as leadership, teamwork, and conflict resolution in real-time, which are critical components of successful collaboration in computing projects [Basholli et al. 2013; Cecilia Bastarrica et al. 2023; Herbert 2018; Hundhausen et al. 2023; Kaluza et al. 2023; Neyem et al. 2014; Vasilevskaya et al. 2014].

Scaffolded Discussion

Scaffolded dialogue is a structured, guided discussion in which a conversation about the team's dynamics, effectiveness, and individual contributions is facilitated. The goal of this method is to encourage reflection on teamwork, identify any challenges encountered, and discuss potential areas for improvement. It provides an opportunity for team members to express their thoughts in a controlled environment and share insights with assessors about how they worked together. Such dialogue can help provide both individual and collective reflection on the team's overall performance, fostering a deeper understanding of the collaboration process [Isomöttönen 2014; Neill et al. 2017].

Panel

A panel is a group of experts meeting with student teams to discuss the progress of their work. The panel members ask the students questions about their progress and provide guidance to improve and complete their work [Tubino et al. 2020, 2021].

Portfolio

A portfolio is a collection of documentation about an individual student's contributions to a project. Unlike a simple log, a portfolio may include reflections, detailed descriptions of tasks completed, and evidence of problem-solving approaches taken. Portfolios help assess both the quality and scope of an individual's contributions, offering a comprehensive view of their engagement in the project. This assessment method also encourages students to reflect on their learning and the skills they've developed throughout the project. Portfolios are particularly useful for highlighting personal growth and achievements in a team context [Allison et al. 2024; Isomöttönen and Ritvos 2021].

Interview

An interview discussion involves a meeting with an individual or the entire team to gain a deeper understanding of their contributions to the project. This method is helpful in gaining detailed insights into how each team member worked on the project, their problem-solving processes, and their interactions with others. Interviews enable more nuanced assessments of individual contributions and can help determine whether the project was a true team effort or if certain members carried more weight than others. It is particularly useful for clarifying any discrepancies between peer evaluations and actual contributions [Porquet-Lupine and Brigham 2023; Vasilevskaya et al. 2014].

Postmortem

A postmortem involves managers conducting individual and team meetings at the conclusion of each Scrum sprint. In the study, managers were seniors who provided leadership and guidance to the teams. For underperforming teams, they offered moral support and resolved any conflicts within the team. During postmortems, the seniors shared their experiences to help the teams succeed [Christov and Hoffman 2019].

5 Interviews Analysis

Figure 6 shows a representation of the themes and sub-themes identified when coding the interviews described in section 3.2. Naturally, the use of a script to guide the semi-structured interview process has influenced the themes formed in the responses of the interviewees with six of the seven top level themes linked to a question asked. The theme broadly categorised as teaching (or teaching process) was not based on a single question, but has sub-themes that emerged during the discussion of many different questions. Similarly, many of the sub-themes formed elsewhere in the map naturally emerged in conversation.

5.1 Demographics

In total 18 interviews were conducted for a total runtime of approximately 22 hours. Interviews were carried out by different members of the working group and interviewees included 8 members of the working group as well as a further ten participants invited through personal and professional networks. Interviewees were employed by thirteen different higher education institutions, all of which were public schools, with 11 of the thirteen granting degrees up to the doctoral level and two granting degrees only to master's level. Interviewees were based in eight different countries, spread across Europe (7), North America (6), Australia/Oceania (4), and Asia (1). Interviewees' aliases A through R follow all quotes. Instructors' location and experience in coordinating group projects are outlined in Table 7.

Fifteen of the eighteen interviewees completed a survey before being interviewed. Figure 7 shows the breakdown of some of the responses to the questions asked. These questions, from left to right

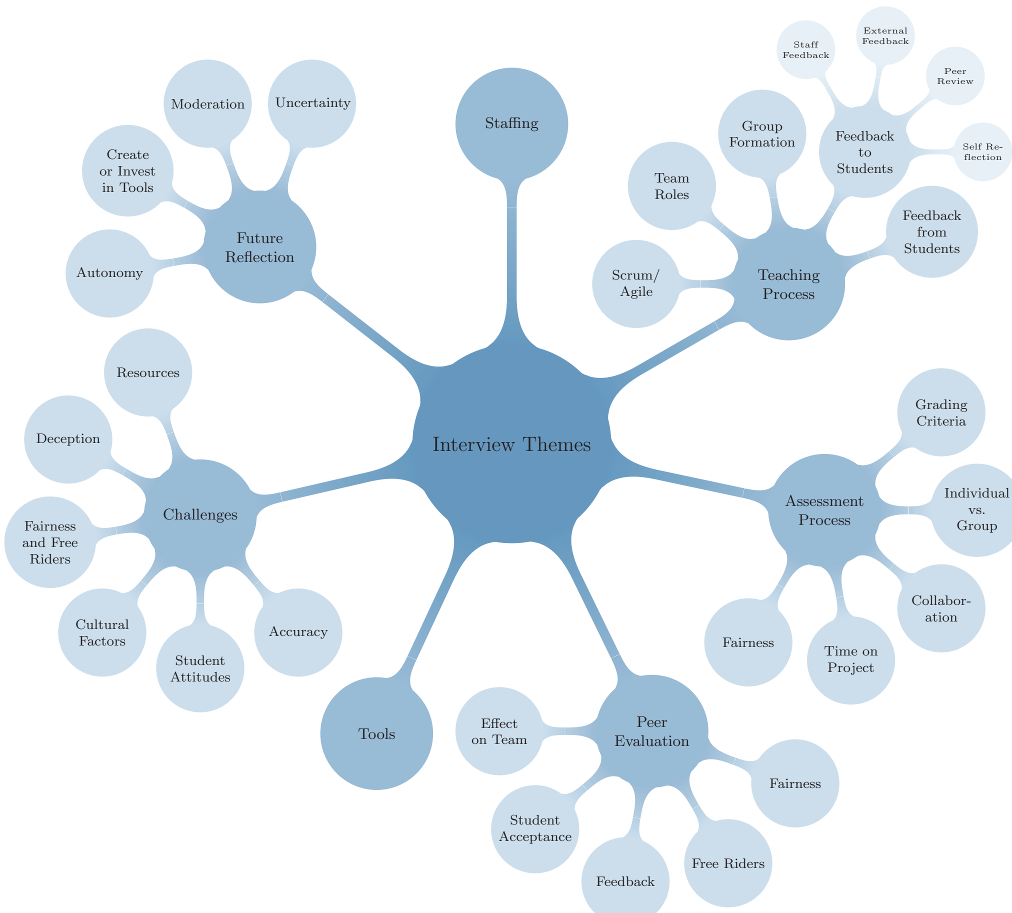


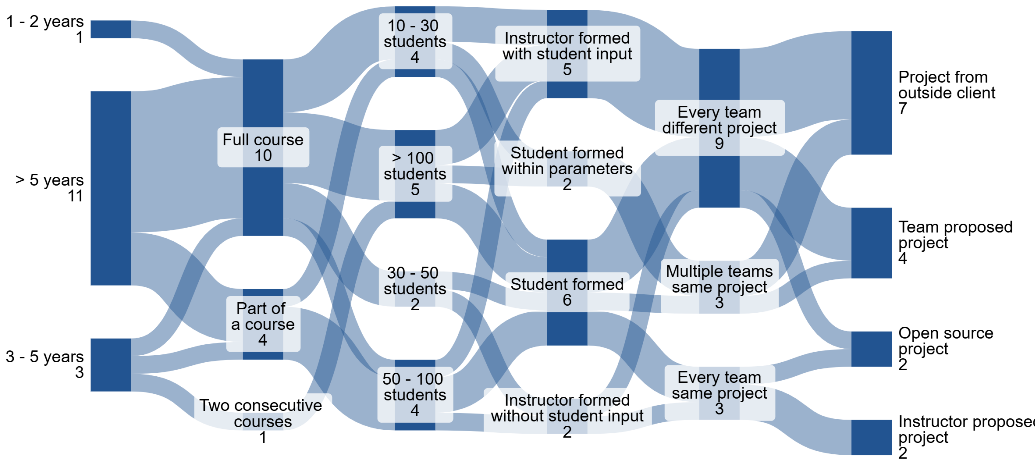
Fig. 6. Map of Themes Identified From Interviews

Table 7. Interviewees Alias, Location and Years of Experience Coordinating Group Projects

Alias	**Location**	**Years**	**Alias**	**Location**	**Years**	**Alias**	**Location**	**Years**
A	England	5+	G	Australia	5+	M	USA	3-5
B	USA	3-5	H	New Zealand	5+	N	Sweden	5+
C	Australia	3-5	I	Canada	5+	O	England	3-5
D	Netherland	5+	J	Canada	1-2	P	New Zealand	5+
E	Canada	5+	K	Scotland	5+	Q	Germany	5+
F	China	5+	L	USA	5+	R	Germany	5+

in the diagram, were their years of experience, the significance of the project they spoke about, the number of students in the class, how the groups were formed, as well as how projects were distributed and the source of the projects.

Fig. 7. Diagram showing interviewee survey responses



Given the number of responses, it would be difficult to use these answers to establish trends or connections. However, the data seems to show a lack of a connection between the size of the class and the method used to form teams. For any of the four class size classifications, no more than 50% of the interviewees reported using the same method of group formation. There does appear to be a relationship between the access to an outside source for projects and the requirement or allowance for groups to complete different projects. Of the nine interviewees that reported allowing each team to complete a different project, 5 of these sourced projects from an outside client and one made use of open source projects, the remaining three allowed the teams to propose their own projects.

5.2 Staffing

Levels of staffing for the purposes of assessment ranged from situations where the interviewee was the only staff involved in the course, to groups of staff members collaborating that were small (2-5), medium (6-15) or large (16+) in size. Multi-staff setups could include team supervisors, coordinators, other faculty, teaching assistants/tutors, PhD students, IT staff, product owners, and external clients.

Training of assessment staff was similarly varied. Some interviewees reported relying only on general training provided by their universities for staff or assistants. Other interviewees had more specific training for group project based courses that was done with the help of meetings, oral or written instructions, and via rubrics. Assessment setups varied from one person assessing all teams, multiple graders assessing multiple teams, pairs of graders assessing every team, to a separate supervisor assigned to assessing a single team.

5.3 Teaching Process

While the teaching process is technically outside the scope of this work, it is intrinsically linked to assessment, therefore we discuss it for context.

The teaching processes described by the interviewees varied based on factors such as the level of the class, the desired learning outcomes, the majors of the students, and the involvement of external entities. Thirteen of the 18 interviewees described a process based on the Scrum or Agile methodologies. This included the appointment of scrum masters, daily or weekly scrums and other regular scrum ceremonies, analysis of risks and blockers, sprints, requirements analysis, and backlogs.

5.3.1 Team Roles. The approach to the use of roles within student teams was also quite varied. In multidisciplinary courses, such as game development, team roles were required as a practical consequence of the project being undertaken. Most interviewees described a flexible approach towards roles within groups wherein students were not required to take on specific roles but could self-organize this way if suitable. Alternatively, interviewees may choose to lecture on the topic of roles within software development teams, but not enforce these roles on the students. One course required that each member of the group take on the role of scrum master for at least one of the sprints completed during development.

Roles in projects

> [M]inimum of four people and you needed to have at least one programmer and one artist (H)

> We don't force students to take on those leadership roles. They should try and elevate themselves to that position if they feel that it fits their group dynamic. (O)

> So I have a few lectures on how to do group work. And show the importance on how to work in a team. [...] I talk to them about Belbin team roles [Belbin 2010]. (P)

5.3.2 Group Formation. In a class, the formation of groups of students is a task that can be difficult to complete in a manner that does not advantage or disadvantage some of the students [Forshaw et al. 2025]. As such, this was a topic that was discussed by many of the faculty interviewed. A common approach to team formation was allowing students to self-select into teams either partially or wholly, or simply to allow them to have some influence on team placement. An example of partial self-selection would be grouping students based on the projects they have declared an interest in. Cases where some influence is allowed would typically involve allowing students to specify someone they would like to work with and students they request not to work with.

Student influencing team formation

> I ask, "is there anyone that you'd like to be paired with?" And I say "I'll try to accommodate that." And then I say, "is there anyone you don't want to work with?" And I will absolutely accommodate that. (B)

Other approaches discussed include randomisation, partial randomisation, pairing weaker and stronger students, and ensuring acceptable gender balance in teams. Several interviewees described completing the process themselves so that they had control and could avoid some potential problems that they had experienced previously.

Instructor control over team formation

> I put the teams together so I have a certain amount of control over what happens (B)

One interviewee described a process they had developed themselves whereby all students were asked the grade that they aspired to get in the project and then used this as a guide to the team formation process.

Aspiration based formation

> In the beginning one of the things we ask for is the grade aspiration and then we try to form the groups according to similar grade aspirations. Then we randomize the groups, but for people with similar grade aspirations. (N)

5.3.3 Feedback to Students. One aspect of the teaching process highlighted by many of the interviewees was the importance of giving proper feedback to the students.

> Importance of feedback
>
> Give students feedback. So it's not enough to say "okay, this is your grade." It's "this is why you got that grade." This is the categories in which you lost points, [e.g.] you didn't volunteer for tasks. You did all the tasks that you were assigned, but you didn't volunteer for any. (B)

Types of feedback given to students was again varied. Typical examples include feedback directly from the course instructor/coordinator or team supervisor, feedback from other students through peer-review, and feedback in the form of self-reflective exercises.

In cases where the group project course includes involvement of external companies as the source of potential projects an extra avenue of feedback is available. The manner of this inclusion varied from cases where the industry partner met with the project group weekly, had a set number of checkpoints during the course or simply provided feedback at the conclusion of the project.

> Feedback from external clients
>
> I think it was a really great learning experience for them because they were all sort of supporting each other and upskilling and the technical areas that they're a bit weaker on and the industry project sponsor was amazing as well and that they had weekly meetings with them and they were given a lot of advice on what they were doing. (H)

Having the students reflect on the work they have done is less direct than the other forms of feedback mentioned, but can help students understand what they have done well and poorly. Interviewees discussed approaches to this that were quite structured, such as reports to be individually completed, reflective essays, and more formal surveys.

> Reflective exercises
>
> I've been using a rather formal approach: it's a rubric that I'm sending out to the students at the very end of the semester usually after the very last deliverable is turned in and it has two parts to it. The first part is three or four open-ended questions where I'm asking each student individually to describe their own contribution, like if they could label their primary responsibility (L)

5.3.4 Feedback from Students. Several interviewees admitted that in their group project classes they didn't formally seek feedback from students regarding their grading process. Instead feedback was, in the vast majority of cases, informal. With many of the students engaging in dialogue with instructors, being largely concerned with their own grade, as a result of the grading process. Based upon the informal student feedback, instructors have been selectively refining their assessment practices. One interviewee stated that they developed their group working strategy with direct input and assent being provided by student representatives. Another drastically altered their grading, pivoting from a shared group grade strategy to an individual-based approach aided by peer review. This was in direct response to feedback from two teams, highlighting frustration at carrying a free-rider to a high grade. Indeed, many of those interviewed highlighted similar complaints from their students regarding non-contributors and the grading process.

All interviewees who mentioned the formal end-of-semester course evaluations said that those do not contain any actionable feedback related to assessment in particular.

5.4 Assessment Process

Interviewees outlined assessment approaches that were different in their structure and execution but had significant overlap along a number of categorical dimensions outlined below.

5.4.1 Grading Criteria. The interviewees discussed grading criteria that varied widely. This is potentially a consequence of courses with different contexts, such as game development or software engineering, as well as different goals for what the students should learn. In some cases the goal of such a project might be for students to learn and apply a methodology like Agile, in other cases it might be learning technical skills such as modelling, while in others the familiarisation with more soft skills might be desired.

Further complications can be introduced when projects are completed in collaboration with outside entities. One interviewee described being unable to directly assess the output of student groups because the code or repository contained proprietary data. This results in grades that are associated with ancillary documents as well as peer and self evaluations.

The specific criteria can most easily be split between skills or outputs that are technical (Product), organisational or more general soft skills (Process + Professionalism / Progress). Interviewees described organisational criteria based on the overall process, the students' use of Trello boards or burndown charts for organisation as well as through the production of project plans or project production plans. Technical criteria mentioned by interviewees included the output or resulting software produced, design documentation from different stages of the development as well as the correct use of tools such as version control, issue tracking, or CI/CD (continuous integration/continuous delivery). Finally, a number of interviewees described criteria for soft skills such as teamwork, communication, presentation, compilation of meeting minutes, team retrospectives, and preparation of reflective reports (Progress).

5.4.2 Individual vs. Group. All of the interviewees reported that they currently awarded individual grades to students, rather than a shared group grade.

Interviewees described a number of different approaches to the distribution of grades across the members of the group. In some cases, individual grades were determined by an individual report completed about the group activities. A common approach to determining individual grades was to use peer-evaluations or contribution statements to scale the group grade for individual students.

> **Scaled grading**
>
> Which means that even though your whole group got like 100 as their group mark, but if your individual mark equals to zero, then zero times the group mark, you still get zero. (G)
>
> ---
>
> We have a maximum variance of 30%, so you might end up with a 1.3. I might end up with a 0.7. (H)

In some cases, interviewees artificially limit the effect that this scaling can have on the individual grade of a student. This can and does lead to situations where a student failing to contribute to the project, could end up with a passing grade as a result. Others described a more holistic approach where the results of peer-feedback and other metrics were gathered and used as a guide to help determine individual grades.

> **Holistic distribution**
>
> So I use [scores from peer-review and contribution statements] as a guide to how I'm distributing the points, not a definitive formula that spits out the answer. [...] I consider it in effect that the whole group has earned a pot of points and some students will get more of that [pot] and others will get less. (F)

Most educators agreed that trying to verify individual contributions is extremely difficult and time-consuming and not always possible with the resources and the scale they were working with. However, a consistent sentiment across the interviews was a general striving to have the most accurate gauge of individual contribution that was possible with the level of resources available to them.

5.4.3 Collaboration. Assessment of collaboration was a concern of a number of interviewees. This sometimes presented as a concern over whether students were actively collaborating on their project, or treating the project as a collection of individual tasks.

> Working collaboratively
>
> I try to explain, you know, the whole thing is about a collaborative exercise and trying to get them to work together and then they'll immediately go and divide up separate exercises and say, okay, you do this one, you do this one, you do this one. I was like, no, no, I want you to work together on this, you know. If you are the only one who does the class diagram, then how are you going to know if it matches with all of these sequence diagrams? (F)

Other concerns raised related to the ability to effectively assess the collaboration of the students within their groups. One interviewee discussed the inclusion of assessment based on the students' communication about the project as a measure of collaboration.

> Measuring collaboration
>
> There wasn't any way of measuring whether the students did collaborating on these activities. [...] students were pretty good at siloing themselves in their group work. Because part of our assessment [...] was their communication on Slack. So they had to communicate on Slack. And I monitored that. And that's where we collected and viewed how they exchanged information, what they were talking about and stuff, how were the women involved and treated in these activities. And… Yeah, they were really quite adept at siloing themselves. [..] It was very hard to measure who did what. But under the hood, when you're listening to them talk and you listening to what they're doing, it would basically boil down to like one individual would pretty much do everything. (C)

5.4.4 Time Spent on Project. Two of the interviewees spoke about explicitly tracking the time students spent working on their projects and including this as an element of the assessment criteria. However, in both of these cases, this tracking relied on the students to self-report the number of hours they had been working on the project.

> Students logging time spent
>
> So they have to log all of their hours and also say what they are doing [...] and if people want to have a [grade of] eight or higher, they have to reach a certain boundary there as well. (D)

5.4.5 Fairness. The fairness of assessment in group projects was a major concern of all of the interviewees and became the central theme in all interviews. To elevate the fairness of the process and get a clearer picture of students' individual contributions, instructors employed a number of different approaches including in-person team interviews, observation of team meetings, one-on-one interactions, team presentations, contribution reports, self-reflection, peer evaluation and review, and analysis of version control logs. Even with these varied techniques applied, not a single participating instructor expressed certainty that their assessment strategy is completely fair to all students.

One of the principle concerns with respect to fairness related to accurately grading individual students' contributions to the project. These concerns were amplified when students were not all equitably contributing (or were perceived by peers not to be equitably contributing) to the project. This was an issue especially when teams were mixed and included students that were not writing code but were contributing to a narrative, to artwork, to design, to project management, etc.

Imperfect measures

Then it comes time to marking and trying to find a way that is fair and equitable to ensure that students are getting grades which reflect the effort and the commitment they've had to those projects where, none of the measures we have are perfect. We have burndown charts, we have daily scrum recording, we have peer evaluation we have student feedback on each other, but that only tells part of the story. And so it's trying to find that balance of making sure people and their contributions are accurately reflected in the grade they get. (H)

The worse or negative [student feedback] would be "so and so didn't pull their weight and we all get the same grade." (C)

But I think that there is a kind of difference between the value that a student is adding to a project versus the work that they're doing. And there's a difference between kind of effort. And I would love to see a way of exploring that. (O)

Fairness could also become impacted on teams with gender imbalances, with personality conflict or flawed team dynamics.

Lone wolves

So one of the biggest challenges we have is with interpersonal conflict in teams. And that can lead to kind of lone wolf behaviours where someone thinks they're doing all the work themselves, but then none of it is integrated by the rest of the team. And that, of course, impairs their collaboration marks because you know their work's not actually making it in the final version. (A)

It's hard to reward the people that are really contributing a lot. Sometimes there are people who contribute too much and kind of hold on to it and don't give it, share it with the team. So it's on them too. (I)

One aspect of fairness discussed was in the context of uniformity of grading especially when multiple personnel were participating in assessment. Typically, a lack of resources (time or staff) result in faculty having less familiarity with the status of individuals within group projects.

Team attachment, hard work and incomplete information

Supervisors are also heavily involved with the team and they are probably a bit biased and really love their team and are going to be on the higher side of grading sometimes and not being so objective. (D)

It takes a lot of energy and effort to do this fairly. [..] Do I do good enough? I keep records and tracks of my assessment to make a fair judgment at the end. So it takes a toll to do it and it takes effort. (N)

It's still hard because we cannot see everything and we're not there sitting next to the student every day seeing what they're doing. (D)

However, within the constraints they are working under, interviewees employed a number of methods to improve the fairness of the assessment process and decouple individual marks from the group grade.

Better than the alternative

The total that they get in the peer review [..] out of 100 is then the percentage that I multiply the deliverable value by. That's how they get their scores. [..] Is it 100% fair? No. But is it more fair than it is to just give everybody the same grade? I think so. (B)

One interviewee commented that they did not think it was possible for them to design an assessment scheme that was completely fair. However, they noted that it was the students perception of fairness that actually impacted the students engagement with group projects.

> Student perceptions
>
> It's not really about when I deem [the course assessment] to be fair, it's when the students perceive it to be fair that matters. (F)

5.5 Peer Evaluation

Peer evaluation, or peer review, was a central topic in our conversations with instructors. Even though in the interviews the two terms were used interchangeably, within the context of the five P's the two are different: *peer review* is used when describing students reviewing each other's work (e.g. code), while *peer evaluation* refers to students providing feedback on peers' overall collaboration, performance and contributions. Because of this theme's prominence, we discuss it separately, although there is some overlap with other themes and subthemes like fairness and feedback.

Peer evaluation was used as an influence to or a component of the course assessment by all but two of the instructors interviewed. In one of these cases the interviewee discussed the use of peer evaluation, but that it had no influence on the grade of the students. The motivations behind the inclusion of peer evaluation were many and varied. Given the discussions of fairness and individual grades in section 5.4, this should not be unexpected.

5.5.1 Fairness and Free-Riders. Mechanisms for peer evaluation can be a useful tool when looking to differentiate the contribution of individual students to a piece of group work. This can be especially useful when those contributions are not as easily quantifiable as something like lines of code or image assets.

> Intangible contributions
>
> Students self-report and peer report a lot of intangible contributions, which otherwise wouldn't be recognized if we didn't have a peer evaluation process. (A)

Another motivation is the perception of students, and indeed faculty, that lack of a peer evaluation could lead to unfairness in grade distribution. This can manifest in students receiving grades without achieving the desired learning outcomes or simply coasting by on the work of others. Peer evaluation can allow situations like this to be identified either afterwards, so commensurate grades can be assigned, or during, to discourage free-riding students.

> Undeserved pass grades
>
> [U]ntil I got really annoyed that students are passing that shouldn't be passing. [...] So then I thought, okay, I'm going to throw in a peer review. And I'm going to make sure that students fill out the peer review. And that peer review is going to influence their peers' grades. (B)
>
> ---
>
> [B]ut also it's like Damocles' sword right? If you don't do well, your peers will say so. So there's this little threat of, yeah, maybe I should contribute. (B)

Most of the interviewees conducted peer evaluation through confidential surveys or individual reflection reports. Four of the instructors conducted live peer evaluations utilizing in-person meetings with the team. Peer evaluations were used at multiple checkpoints or at a single checkpoint at the end of a project. Interviewees who didn't use peer evaluation at all or only used it at the end of the course expressed a desire to utilize it more often but were precluded from doing so due to the lack of resources and time they'd need to devote.

More peer evaluations

I'm starting to think that I should do a team self-assessment at every stage of the project. I'm starting to feel that that would be a more fair way to do things. Because I feel like that would create more accountability for the students that are not putting in this much work. (M)

I think what works for me is [..] telling students from the very, very beginning and reminding them throughout the entire course that that peer assessment of teamwork is coming. That it is a part of their work that they that they will be held responsible. Now, having a formative aspect to it would be also very helpful. But in some ways my reminders about this.. I'm not saying it replaces formative assessment, but it helps, I think. And I think it helps keeping students aware that they shouldn't be freeloaders. (L)

5.5.2 Feedback. One aspect of peer evaluation was the decision to release (sanitized) comments made by teammates to the students they were made about. The interviewees were split in this regard. Some purposefully did not release the feedback either due to negative past experience, cultural considerations, or due to simply not wanting to accidentally negatively affect team dynamics. One interviewee explained that they did not want to release the comments that student have made about their peers as they didn't have time to check for offensive or inappropriate statements that would cause problems in the group. Conversely, others released the comments to the students they were made about. Other interviewees said they actively encouraged the practice of sharing peer evaluation derived feedback with the students, though in some cases this was limited to the positive comments made about the student and suggestions for improvement.

Consequences of peer comments

Basically, I don't want to cause problems, cause drama. (L)

[M]ake sure that if somebody said something good about them to tell them that because that boosts morale like nobody's business. (B)

5.5.3 Student Acceptance and Effect on Team. When discussing how the students feel about peer evaluation, the general consensus was that students were in favour of the practice.

Being heard

I think most students were relatively happy to to engage in the process because they saw it as an opportunity for them to address any issues that they might have had. (H)

We have had some positive feedback on the peer evaluation process. I think it's on the basis of they like being seen and having a voice. (A)

And I think students being able to voice their opinions anonymously is also quite useful. So that if there is an issue, it's not just everyone patting their back and pretending that everything's okay. (A)

Several interviewees reported that student feedback on having peer evaluation outcomes tied directly to grades was negative. Consequently, the activity was continued as a source of information about the overall health of the groups or an indicator of potential problems that could be addressed directly by group supervisors.

Direct link to grades

No, it doesn't directly influence the marks and that is something students have fed back to us through surveys. They don't like it when it's directly linked to the assessment process. (A)

> I would say it [peer assessment scores] shouldn't be used directly [for grades]. So what I personally I'm still figuring it out myself. But what I currently do is that I look at it and then it's first of all, it's a warning sign. So that's kind of like you need some evidence when you want to make a decision, right? (J)

Amongst the concerns raised were the possibility of creating a toxic environment, the use of peer evaluation as a punishment, and the use of peer evaluation based on personal feelings.

> **Misuse of peer evaluation**
>
> Yeah, at the risk of sounding critical, I think it perhaps was implicitly and unfortunately, sometimes possibly explicitly suggested to students that things like peer evaluation can be used to punish group members or used as the threat of punishment of group members. It's like, if you have someone who isn't contributing you can say to them, remember there's a peer assessment and you'll get marked down if you don't contribute to that, which I think is absolutely the wrong idea. It's creating competition within groups effectively or you know perhaps creating like competitive or toxic environments. (H)

Each of these has the possibility to negatively effect the dynamic on the team and consequently the work they have to produce.

5.6 Tools

Interviewees identified a number of tools that they were either currently using or were considering for use in the future. These tools can be grouped into a number of broad uses, general tools such as learning management systems (LMS), tools for communication, tools for organisation, tools for administration, and tools for managing peer feedback.

Slack[3], Microsoft Teams[4], Mattermost[5], and Discord[6] were frequently discussed for enabling intra-group communication. These tools were a topic that encouraged a certain level of debate regarding suitability for purpose. Multiple interviewees put forward the idea that the platform Discord was too familiar to many students as a social platform and as such its use was more likely to result in unprofessional language and behavior during group discussions. One interviewee gave advice that regardless of what technology was being used, it should be within the university IT system.

> **Communication tool advice**
>
> There's been accusations of malicious communications through those online platforms. Which is very difficult for us to intervene in if it's on a social media because it's considered kind of outside of our jurisdiction but if it's on our platform, then it is within our jurisdiction and then we can act upon it accordingly. (A)
>
> ---
>
> I tried to get rid of a Discord platform at the University of [redacted] because it was the 'Lord of the Flies' in there. (C)

[3]https://slack.com/

[4]https://www.microsoft.com/en-us/microsoft-teams/group-chat-software

[5]https://mattermost.com/

[6]https://discord.com/

With respect to organisation, almost all interviewees mentioned the use of Git[7], GitHub[8], GitLab[9], or more generally version control systems. The use of these tools to aid the distribution of grades was also mentioned, though it was never explicit.

The number of commits made by individuals was sometimes used to check students were contributing but instructors were aware of the ways students could use deception to appear as if they were doing more work than they actually were. One interviewee reported using the tool Gource[10] [Caudwell 2010] to visualise the contributions of the students to a repository over time. Additionally, analysis of logs such as git-blame were used to evaluate the contribution of individuals to the deliverables in assignments.

Trello[11], Jira[12], and Microsoft Planner[13] were mentioned for task tracking. One point of concern raised was the reliance on tools and platforms that can suddenly become unavailable or too expensive for use in an educational context. Trello and Slack are both examples where a change in pricing or terms of use can limit the utility or availability of a tool currently in use.

Tool availability
I don't know what we're going to do about Trello now. (B)

The most commonly mentioned tool for administration was Microsoft Excel[14]. In some cases, it was used to gather and combine disparate results from the course into final grades. In other cases Excel was directly used by students to record daily scrum logs and to enable tracking of work in the form of burndown charts.

The management of peer feedback within software project groups was relatively varied. Solutions such as paper-based forms, Microsoft Forms[15]/Google forms[16], FeedbackFruits[17], CATME[18], and SPARKPlus[19] were all mentioned as being currently in use or something that the interviewee was considering for use in their classes.

One participant described a custom built assessment system that integrated with their LMS and git server. This system allowed them to manage individual and group submissions as well as linking them to tags in their repository for review.

Several interviewees also mentioned large language models/artificial intelligence (AI) but they were still uncertain exactly how to use it for assessment of group projects.

AI
I think certainly AI can be useful, I still don't know how it can be useful for this very specific purpose. (K)
And then for scaling the courses [..] we could start using or be thinking about using some AI techniques or LLMs to get feedback on the group performance. To see which parts should we look at, what are certain outliers, etc. To just scale the grading process better. (R)

[7]https://git-scm.com/
[8]https://github.com/
[9]https://about.gitlab.com/
[10]https://gource.io/
[11]https://trello.com/
[12]https://www.atlassian.com/software/jira
[13]https://planner.cloud.microsoft/
[14]https://excel.cloud.microsoft/en-us/
[15]https://forms.microsoft.com/
[16]https://workspace.google.com/products/forms/
[17]https://feedbackfruits.com/
[18]https://catme.org/
[19]https://sparkplus.com.au

5.7 Challenges

Not a single one of the interviewees expressed full satisfaction with their assessment strategy for group projects. Smaller challenges were outlined such as limitation imposed by institutions, the difficulty of achieving uniform grading, the lack of agreed upon best practices and dealing with accommodations and extension requests in group work contexts. The larger challenges are addressed below in the following sections, in answer to "**RQ2**: *What challenges do undergraduate computing educators encounter when assessing student group projects?*".

5.7.1 Accuracy of Grading. Some instructors felt that their assessment process and criteria were more holistic and hard to exactly translate into quantifiable grades. Others considered theirs to be overly prescriptive and constraining because of the use of rigid rubrics that stifle student creativity. Interviewees felt that when students worked on different projects, it was difficult to keep grading criteria equal among groups.

> Different project scope
>
> It's very difficult, though, because the scopes vary so much. How do you equalize things? (F)

For courses where a supervisor was in charge of multiple graders, and graders only worked with a single team or a few teams, objectivity was an issue.

> Grader objectivity
>
> What's also really hard is that the supervisors are also heavily involved with the team and they are probably a bit biased and really love their team and are going to be on the higher side of grading sometimes and not being so objective. (D)

The most often cited challenge by all interviewees without exception was the successful separation of individual from group contributions for assessment purposes.

> Individual vs. group contributions
>
> I believe that the individual contribution to the project, that's the most difficult one. (E)
>
> It's very hard to see every individual contribution if you're not there all the time. (D)
>
> As in all team projects for students it's difficult to identify the high performance students and not so much motivated elements of a team. (Q)
>
> Most challenging in assessing group projects is the attribution, I would say. (R)

5.7.2 Student Attitudes Toward Group Work. Some interviewees expressed that getting students to collaborate instead of siloing themselves or others off, creating hierarchies, or splitting into subteams that did not communicate was a challenge.

> Getting students to collaborate
>
> For me the most challenging part was actually getting [students] to collaborate. To view it as a single project instead of a collection of tiny little pieces of work that would happen to be side by side in a repository. (F)

5.7.3 Cultural and Socio-Economic Factors. Diverse teams have the potential to surface insensitivities and prejudices, which disrupt mutual expectations of how team members ought to conduct themselves. Social norms differ for various reasons and might not come from a place of malice but of a lack of experience. Sometimes, unfortunately, these facets influence the assessment process in ways that concern educators, especially if they ostracise members of the student community.

Equity

> Occasionally we do observe behaviours that would be considered a breach of [equality legislation]. Sometimes it's quite egregious and [..] sometimes it's kind of insidious. That interacts with the assessment process, particularly if people feel disengaged or estranged and they don't feel able to contribute. So that's one of the challenges we're still grappling with. (A)

Similarly, internationalisation needs to be managed with care. Multilingual cohorts might prefer to work in their own language, creating friction with tracking and isolating team members from different backgrounds, which would affect how they perform on any collaboration assessment component. For non-native speakers, the language barrier may create issues around communication, especially where interpersonal communication is assessed.

Language barrier

> Often it's just the technical jargon; that they're not familiar with the technical jargon because they're trying to translate it in their own language and then into English and it means that they're not as responsive or they're not as quick to communicate. (A)

> I certainly observe that in our [..] students. They're visibly frustrated when they're in teams with international students who find it difficult to communicate in English quickly. And conversely, [the international students] often find it difficult to contribute, [they] report to me that they often find it difficult to meaningfully contribute to things like supervision meetings and studio practice because the other members of their team are speaking too fast. (A)

> I heard stories of some domestic students being pushed out of their groups because the international students said, oh, they're not working with us. And it turns out because they're speaking their native language, they excluded the domestic student. (C)

An important consideration is cultural diversity. With an increasing trend in the use of peer evaluations, it becomes clear that these differences diffuse into the assessment process. One instructor expressed surprise that some of their students, who are culturally very direct people, refuse to evaluate their peers because they don't feel comfortable passing judgment. This is especially the case before the end of the semester when they still have to work with the same teammates. These same students had no problem discussing their own contributions to the project. Another interviewee had to have a conversation about realistic expectations with an international student who was overly critical of their teammates and, during the first peer evaluation, failed to assign the highest mark in any of the categories for any of their peers even though that mark would have been well deserved.

Cultural attitudes toward peer evaluation and self assessment

> I noticed that certain students from a particular culture are more inclined to provide feedback and about their peers. [And] self evaluation is not considered a good thing in certain cultures. So students don't like to say, oh, you know, I contributed 100%. They do not want to, you know, to self evaluate. (E)

> I think because of culture background, some students feel that they would be very rude if they point out something in front of their [fellow-]students in the class. So they prefer to hide and wait until the end. (G)

> A student like a student from [redacted], they feel that it is rude if I report anything wrong for my peers. (G)

In terms of student acceptance of grading, one instructor said that in their opinion, female students tended to be more accepting of their grade, whilst male students could be more vocal and likely to challenge a grade. It was also noted that in groups with gender imbalances, students could be relegated to lesser roles by their team or by self-isolation, which would affect the way in which they collaborated with peers.

Group gender imbalances

Likewise, when there was one woman in a group I also noticed that they would be very hesitant to contribute to the project. And so how do we make that fair? How do we help them because as a result, they might not have, they didn't get the high marks in the chat and the communication because they were apprehensive to put input in. So that also was difficult in fairness. (C)

Other interviewees claimed similar issues across national lines. Students from cultural backgrounds with high power distance and respect for authorities seemed reluctant to question assessment practices. Additionally, it was noted that students from some cultures are highly grade motivated, with corresponding anxiety about how they would demonstrate exemplary performance in a context in which they lacked full agency.

Socio-economic factors may also create friction and inequities in contributions among team members. Students who are not able to attend all team work sessions or have less time because of having to work to support themselves may be perceived by their teammates as free-riders. This is often discussed in relation to changing conditions for students and the general increase in the cost of living.

Socio-economic factors

Commuting in is often a challenge for our students and does often create frictions in teams where people prefer to work online rather than come in because of the difficulty of commuting. (A)

There is a non-trivial proportion of our students which don't sympathize with our poorer students as much as we imagine that they ought to be. (A)

And certainly we've had frictions even between academic teams and students where they sign up for a full-time course. We expect them to come in and, well, we expect them to contribute to the team. But if they've taken a full-time job because they can't afford their rent then that creates a kind of tension between supervisors and the students that they're supervising. (A)

5.7.4 Fairness and Free-Riders. All interviewees were worried about fairness and free-riding. Peer evaluation has emerged as a trend in striving to address these challenges but it comes with its own set of issues and has the potential to erode team dynamics.

As mentioned above, instructors were not convinced of their ability to separate and assess individual contributions from the group work and about ensuring free-riding was discouraged, detected and sanctioned.

Sharing grades and free-riders

Students really don't like sharing the same grade. Students get very upset, even if it's just a small component of the overall module because they see it as their peers getting credit, which is undeserved. Particularly if they perceive them as being a free rider. (A)

Our students are very critical of each other and they often say we're too easy to pass people who really shouldn't be here. (A)

I've had students before get very, very vocal, as you would expect, when other students aren't pulling their weight. So they can very, very clearly say, hey, I think this person's done no work. They have never shown up. I've never seen them. They don't commit anything, they don't trip

> to any of the meetings. [..] Are they going to pass? Are they going to pass off of our work? Like, how dare they? (O)

Some interviewees were bothered by students trying to artificially inflate their participation by overemphasizing their contributions in structured self-reflection activities. One interviewee who split the team grade into non-overlapping individual grades based on individual contributions and took into account attendance, peer evaluation and Git logs to determine the proportion each student received, said that this strategy had helped convince students that they will receive fair grades, even if one of their teammates stopped contributing as their team score would be split fewer ways. Additionally, students were discouraged from overinflating teammates' peer review scores or they would receive a lower grade as a result of the split. This approach made it easy to spot free-riders.

5.7.5 Deception. Naturally, related to fairness and free-riding, during the interviews attempts at deception were also cited as an issue. Cases included students covering for their absent or free-riding teammates (because they were friends, because they had agreed to cover for one another, because they didn't feel comfortable telling on each other, for cultural reasons etc.) by submitting inflated peer evaluation scores, entering absent teammates' names on attendance sheets etc. In other cases, students tried to circumvent communication rules by using unapproved channels, or to deceive by falsifying work.

> **Student attempts at deception**
>
> So there's an attendance sheet done. I forced them to sit with their group mates when they're in class. [..] So I could basically come up and say, okay, there's five of you there, but six names [on the attendance sheet]. (F)
>
> And I noticed one student was being really cheeky, he said, "type a lot in the chat room so it looks like we're talking a lot" and I jumped in and I was like, "no, it needs to be meaningful". And he's like, "oops". (C)
>
> I'm just going to commit a bunch of empty lines to show that I did something. (C)
>
> To judge individual work. Sometimes students are not honest with me. Sometimes they cover one of their colleagues who didn't contribute. (E)
>
> Based on the statistics we collected, we can identify that some of the students, they tell the truth about their feelings about the peers. But we can see some of the other students probably just give very high marks to all the peers. (G)

Sometimes students didn't even have to try to intentionally deceive. Being overly punitive to peers, culturally imposed feelings of unease at openly criticizing others, or mere apathy could all result in inaccurate feedback about teammates' contributions.

> **Free-riders, apathy, inattention, and being too nice or nasty**
>
> Sometimes I know for a fact that a student is a freeloader and the team just doesn't care. (L)
>
> [E]ven though we try our best to avoid any free rider, but I believe for each semester we have some free riders hiding in the teams. So if their peers didn't really complain from the teaching team's perspective, we never know that. (G)
>
> Because especially during meetings, a lot of students, we sometimes refer to as kind of [redacted] niceness, where it's like, oh, this student is terrible. And they'll go, yeah, you know, they're doing their best, they're really trying. And that's not particularly helpful. So they can be a little bit more honest. (O)

> All of our teams are multidisciplinary. [..] We find that a lot of the concerns raised by teams, particularly in relation to kind of free riding and other such worries are actually people not paying attention to what their peers are doing and assuming they're not doing anything. (A)
>
> I added another factor in there. So in fact, that project contribution percentage is a combo of what students think and what I think. And yes, students might punish somebody for no reasons. I've seen that happen. (L)

Interviewees named deception and students being overly punitive to teammates as reasons why they used peer evaluation for purely informational purposes. The one instructor that used the peer assessment results to directly assign grades to individual students said that they still reserved the right to override the results based on their own observations of the team and did so when necessary.

5.7.6 Lack of Resources. The majority of interviewees volunteered that they didn't receive enough resources and support to either innovate, make incremental changes to assessment or to even conduct proper group project assessment in their current courses. Lack of time, other supporting personnel, and big class sizes were the biggest culprits.

Some participants wanted to add more assessment checkpoints but were unable to do that because they didn't have the time to devote to that additional workload. Some were dealing with large class sizes as a solo instructor or with insufficient support from graders and team supervisors. A few of the interviewed instructors expressed the desire to try out new tools but they were prevented from doing so by lack of time and/or resources and the lack of technical support from their institution.

> Scale and lack of resources
>
> I think if more resources or more support can be given from the institution, that will be highly appreciated. (G)
>
> Scale been a real challenge for us [..] and keeping things coordinated and everyone on the same page. It's not insurmountable problem, but certainly in terms of keeping things organized is quite trying. [..] That can be a challenge just dealing with the scale of the number of staff involved. The other thing is the kind of moderation and just the sheer number of projects and students that we have to moderate. So towards the end of a study block, it is a kind of barrage of challenges. (A)
>
> I don't think that there is anything that is completely fair in this life. But this is right now, the best solution that I can see, [that] I can actually manage. I think there are better solutions, but I cannot manage them because of lack of time and support. (L)

5.8 Future Reflection

To encourage participants to think about future strategies, we asked interviewees to share their perspectives on opportunities for improving the assessment of group work in computing courses. We also invited them to reflect on specific methods or tools that they might like to explore and to consider any potential barriers to implementations. Through these discussions, we identified four key areas that offer valuable insight.

5.8.1 Autonomy. One instructor highlighted the effectiveness of providing students with greater freedom in their projects. They suggested that by outlining the approximate scope of the task, and allowing students to choose a concept that is personally meaningful or familiar to them within the team project to work on, students in turn will build projects around something they are passionate yielding positive outcomes.

Autonomy

I think one of the things that works really, really well for me is when you can give students a lot more freedom. So if you can tell the students the approximate scope of the thing you're looking for and let them go off and pick a concept that's dear to their heart that they know well and have them build the project around that. That can go down really, really well. (F)

5.8.2 Create/Invest in Tools. Unsurprisingly, many instructors had strong opinions when asked about the tools they would use for assessing group work. One instructor highlighted the benefits of a dashboard that provides an overview of hours worked, tasks completed, commits made, and individual contributions to the codebase. Others expressed interest in tools such as SPARKPlus specificity tailored for group evaluation, although they noted some challenges in supporting the implementation.

Some suggested integrating software development tools like version control systems and issue tracking platforms, such as Git and Jira, to evaluate individual contributions and enhance students' understanding of the development process. Suggestions also included utilising software development tools, such as version control systems and issue-tracking platforms like Git and Jira, to evaluate contributions and teach students about collaborative workflows. It is noted that instructors also expressed curiosity about tools used in other institutions, emphasising the need for institutional support and resources. Additionally, the idea of combining quantitative metrics with qualitative assessments emerged as a promising approach to enhance evaluation practices.

5.8.3 Moderation. With respect to moderation of the assessment process, some instructors suggested that grading a single team should involve multiple assessors to ensure fairness and objectivity. Another instructor recommended a collaborative grading approach that should be done in moderation with the person responsible of the course until the grader builds confidence and expertise in their assessment.

Moderation

As he got more experience and sort of got, I guess, a bit more of a feel for where I sort of see the various grade boundaries, he started doing more of the teaching, more of the assessment as well, but it was always a mediated response in a mediated grade in that he would mark his own pieces or he would mark with a piece of assessment, I'd mark all their pieces of assessment and then we would discuss, deliberate and sort of find a grade that we were both comfortable with which would tend more towards myself being the academic, the course coordinator and kind of having responsibility in there. (H)

5.8.4 Uncertainty. The instructors highlighted their uncertainty regarding assessment practices, with one instructor acknowledging their lack of knowledge to confidently change their current approach while recognising the need for improvement, and another acknowledging that there is still significant work to be done in this area. A recommendation was to create a play-book of best practices for instructors to follow, or to make students more accountable by creating team chapters from individual group members to reflect their contributions instead of one final report.

Uncertainty

I don't know. I don't know enough to change my current approach, but I know that it needs more. (B)

[F]rom my perspective, what I'm trying to do, and I don't think I have, I think I probably still got a lot of work [to do] trying to figure out those less quantifiable things in regards to contribution. (H)

> But we don't have anything to benchmark against and we don't really have that much to look to in terms of other examples of how to do things and things which, you know, a kind of playbook of best practices, which might be useful to know. (A)
>
> Well, I think making a very big complex rubric doesn't always work, it's very hard. (D)
>
> I think there is a lot that could be done in group work assessment, particularly around the fairness element and I don't know what that looks like yet. (K)

Interviewees also acknowledged the inherent difficulties in achieving complete fairness, stating that while assessments can never be perfect, incorporating multiple perspectives and striving for clarity can help.

> Acknowledging Imperfection
>
> We also have to accept that it's really hard to do these things and just do our best to have measures for that and have input from different sides and to talk about this and to have multiple views on it, but in the end, it's human work, and it won't be perfect. (D)

6 Discussion

The findings of this study identified a broad spectrum of assessment approaches for student group projects in undergraduate computing education. In this section, we synthesise the findings from the Systematic Literature Review (SLR) and interviews, suggesting opportunities for future research, and offer practical recommendations to enhance the group project assessments in undergraduate computing programs.

To better understand the Computing Education Research (CER) community's interest in student group projects, we assessed papers on the topic of student group work over the past ten years, as illustrated in Figure 5. The figure shows a consistent interest and an increased publication rate in this area since 2020, with 60% of the papers published within the last five years. However, we cannot determine a clear trend in the CER community's interest in this area based on the publication rate, but broadly, we observe a more consistent discourse since 2020.

Prior research [Sridharan and Boud 2019; Sridharan et al. 2019] mentioned that computing education can implement targeted inventions to enhance peer assessment practices by positively contributing to student learning. Indeed, there have been various calls to reform assessment in the software engineering capstone [Basholli et al. 2013; Tubino et al. 2021], which typically involves conducting projects in teams [Tenhunen et al. 2023]. Although in some cases teamwork has been used in tertiary education to reduce the workload of lecturers [Schulz et al. 2023], our findings suggest that educators endorse nurturing students' development of non-technical and interpersonal skills necessary for the workforce [Allison et al. 2024].

Our findings in the SLR and interviews showed professional methodologies, such as Agile, applied in the assessment approaches. Critiques of traditional assessment tools highlight the need for innovative approaches, such as those aligning with Agile. At Imperial College London, integrating project-based learning and frequent assessments inspired by Agile principles has shown positive results. Weekly submissions and bi-weekly checkpoints have encouraged the consistent effort and minimized last-minute work surges [Chatley and Field 2017; Kaluza et al. 2023]. These strategies reflect a shift from theoretical knowledge assessment to practical skill development relevant to real-world software engineering careers. Other studies used tools commonly used in industry, such as Git [Guttmann et al. 2024], within the group evaluation, also used by the majority of the educators we interviewed.

While our SLR found research exploring assessment technologies or tools, these are underrepresented in comparison to discussions with interview participants adopting manual or ad hoc practices, despite the increasing prevalence of online and hybrid group projects. This is a notable gap when synthesising our findings. For the remainder of this section, we present the gaps between the interviews and SLR within six categories. Section 6.1 describes where in the curriculum we found group project classes. Section 6.2 discusses the learning outcomes and Section 6.3 discusses the assessment activities. Section 6.4 expands upon one of the central themes in assessment approaches, that of fairness and individual contributions. Similarly, section 6.5 discusses the peer evaluation focus in our findings. We conclude with Section 6.6 presenting the tools we found applied to facilitate or support the assessment approaches in answer to "**RQ3**: *How is technology used to support the assessment of student group projects in undergraduate computing?*".

6.1 Curricular Placement

The SLR confirms that group projects are most commonly situated in software engineering and capstone contexts, reflecting their suitability for fostering technical collaboration and industry-aligned project experience, helping students prepare for work in a professional software development environment [Hobbs and Tsang 2014]. This curricular emphasis aligns with previous observations [Steghöfer et al. 2018; Stephenson et al. 2016] and is echoed in the interview findings with the majority of our interviewees teaching group project courses mostly related to software engineering and game development using the Agile method. Though there is alignment in the courses typically using group projects in the SLR and interviews, there is a gap in the application of group projects in other tertiary courses. This suggests a missed opportunity to embed collaborative work more widely, including in specialised or theory-driven computing topics. The under-representation of group projects in such areas may limit students' exposure to diverse modes of teamwork and to the kinds of collaboration found in research, infrastructure, or data-focused computing roles.

6.2 Learning Outcome Prioritisation

Existing studies often focus on describing assessment mechanisms without explicitly evaluating their impact on learning outcomes, leading to a limited evidence base on the effectiveness of different assessment strategies. The limited rigorous evaluation hinders our understanding of which assessment approaches effectively enhance student learning. The literature is fragmented, with few comparative analyses of alternative approaches. This fragmentation makes it challenging to identify best practices that could be widely adopted and limiting educators' ability to make informed decisions on assessment approaches for their students' learning.

Our findings show that collaboration [Clear et al. 2020] is a core component of the assessment approaches we evaluated and is a skill needed in a professional development environment. We know that collaborative learning can help students better understand concepts [Lai 2011], but we found a notable pattern in the prioritisation of learning outcomes across project stages, including non-technical skills such as collaboration. While technical knowledge and deliverables dominate early-stage projects, collaboration skills, project management, and reflective practice become more prominent in later years. This reflects established curriculum design principles but may perpetuate the assumption that teamwork competence develops incidentally, rather than requiring explicit instruction and scaffolding [Lingard and Barkataki 2011]. The conflation of contribution volume with collaborative skill, observed in several studies, underscores this issue and highlights the need for framing learning outcomes clearly to ensure constructive alignment [Biggs 1996] with approaches to assessment.

6.3 Assessment Activities

The focus on assessment approaches gave us us the opportunity to look deeper into how researchers and educators construct group activities. We found assessment modalities remain centred on presentations, reports, and code inspections, consistent with prior work [Bastarrica et al. 2017; Porquet-Lupine and Brigham 2023]. Interview participants discussed focusing on mostly the same areas as our findings in the SLR.

Table 8 shows the learning outcomes/assessment approaches discussed by the interviewees when viewed through the lens of different assessment dimensions: *Principals*, *Progress*, *Product*, and *Process + Professionalism*. Each interviewee is identified by a single letter and the data is presented in such a way as to present a more holistic view of the assessment in the classes that they teach. It should be noted that this categorical breakdown of the data was not one of the original aims of the interviews and thus some of the approaches may have been used but not discussed directly during the interviews.

While fully reliable direct comparisons with the contents of Table 6 are difficult, we found a correlation between the SLR and interviewees' responses in the *Products* and *Process + Professionalism* categories. We found presentations to be the most frequently discussed approach for assessing students' products in both the interviews and SLR. We also found, in both the interviews and the SLR, that peer evaluation was the most common approach to assessing students' processes and professionalism in student groups. Some of the more esoteric approaches we found in the SLR, like duels and panels, were not used in practice by our interviewees. There was also little to no emphasis on exams and individual homework. Instead, our interview participants inspected code, graded documentation, written reports, and portfolios consisting of multiple deliverables when assessing the product; used analytics and structured observation or scaffolded discussions and postmortems to assess process and professionalism. Progress was assessed with the help of retrospectives and reflections and the assessment of principals, where assessed at all, was done through exams and individual presentations. Peer and self-evaluations loomed large.

The SLR found that assessment approaches were often chosen with specific contextual goals in mind. These included improving code quality [Chen et al. 2022], automating grading processes [Chatley and Field 2017], and addressing the issue of free-riding, or non-contribution [Cecilia Bastarrica et al. 2023]. Cecilia Bastarrica et al. [2023] also explored how grading strategies can affect team cohesion. The diversity of rationales for assessment choices suggests that educators' priorities extend beyond academic achievement to include workload management, team dynamics, and student wellbeing. This reinforces the complexity of group project assessment, where pedagogical, administrative, and socio-emotional factors intersect. Despite educators' efforts to emulate industry practices, student satisfaction with group assessment remains inconsistent, with some methods, such as peer assessment, failing to garner universal acclaim among students [Crick et al. 2022], findings also echoed in the interviews.

The increasing adoption of peer assessment and analytics, however, signals a shift towards mechanisms aimed at improving fairness and addressing the long-standing challenge of unequal contribution. While these approaches promise to tackle fairness concerns and offer insights into team dynamics, their effectiveness depends on students' willingness to engage honestly, on the cultural norms surrounding peer critique, and on instructors' capacity to interpret the resulting data meaningfully. As noted by Tafliovich et al. [2015], ensuring that assessment strategies are both equitable and valid remains an unresolved issue in computing education.

6.4 Fairness / Individual Contributions

We observed that points raised by our interview participants aligned with some of the research we found in our SLR pertaining to fairness and individual contributions in student group assessments. For example, our interview participants expressed interest in fairness and justice in assessing students' group activities with *free-riding* being an oft-mentioned concern, a topic previously mentioned in Section 4 by Cecilia Bastarrica et al. [2023] that examined marking peer assessment

Table 8. View of the Interviews Through the Assessment Dimensions

Approach	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	#
Principles																			
Exam				✔			✔		✔								✔		4
Home Exercise			✔																1
Presentation	✔				✔			✔									✔		4
Progress																			
Reflection	✔		✔	✔				✔		✔				✔		✔	✔		8
Retrospective	✔	✔	✔	✔						✔	✔			✔	✔				8
Exam + Portfolio																			0
Portfolio																			0
Product																			
Presentation	✔	✔	✔	✔	✔	✔		✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	17
Build Inspection	✔	✔				✔	✔								✔	✔	✔	✔	8
Report	✔					✔		✔									✔		4
Peer Review							✔		✔					✔				✔	4
Portfolio	✔		✔		✔	✔				✔	✔	✔	✔	✔		✔	✔		11
Documentation			✔				✔			✔	✔	✔				✔	✔	✔	8
Duels																			0
Structured Exercises																			0
Process + Professionalism																			
Peer Evaluation	✔	✔	✔		✔	✔	✔	✔		✔	✔	✔	✔	✔		✔	✔		14
Analytics	✔	✔				✔	✔			✔	✔	✔	✔		✔	✔	✔	✔	12
Structured Observation	✔	✔	✔	✔	✔	✔		✔	✔		✔	✔				✔		✔	12
Scaffolded Discussion		✔											✔				✔		3
Panel																			0
Portfolio			✔																1
Interview																		✔	1
Postmortem			✔							✔							✔	✔	4

as a means of addressing non-contributors, students that do not actively participate in the group learning activity. In addition to the peer assessment aspect, the authors used team-based learning (TBL) principles, a three phase group assessment model to support fairness in assessment. These three phases are the 1) project management, the 2) deliverables for the learning activity, and the 3) individual's contribution. Like Cecilia Bastarrica et al. [2023], the work by Quek et al. [2023] addresses individuals' contributions to group activities by also using deliverables (or, in their study, they refer to them as milestones) and project management for transparency in the group. This work, based on socio-constructivist principles, assesses the individual's contribution through self-reflection exercises. Both of these studies leveraged deliverables over the course of the project and applied aspects of project management to address fairness in the group activities. This raises future opportunities to explore the use of project management and deliverables in group project courses to improve fairness in group assessments, while minimizing additional efforts by educators. Avoiding added workload is especially important. Our interview participants stressed being time-poor in their existing courses and it is not a stretch to assume that this is the experience of other educators as well.

The diverse approaches to evaluating individual students' contributions incorporate elements of peer, self, team, and teacher assessments. Evidence suggests that students' perception of effort is not objective [Herbert 2007]. Crick et al. [2022] introduced the Team Q model, designed for summative feedback, focusing on five core components of teamwork: contribution to the project, facilitating contributions of others, planning and management, fostering a team climate, and conflict management. Team Q was found to support gender parity, as peer assessments were unbiased by either the gender of the assessor or the assessed. This highlights its potential for fair and inclusive evaluation in team-based settings.

6.5 Peer Evaluation

Closely related to fairness and used to decouple individual contributions from the group work product, we found peer evaluation/review was a central topic emerging from our interviews. We also found a variety of peer assessment approaches used in 60% of the SLR papers. Peer evaluation significantly enhances engagement, as evidenced by 58.49% of students reporting they would not have put in the same effort without graded assignments [Kay 2022]. Oral interviews also play a critical role in ensuring accountability, particularly in team settings, by holding free-riding members responsible for their lack of participation [Porquet-Lupine and Brigham 2023], an approach also employed by some of our interviewees.

With peer evaluation commonly discussed in the SLR, our results found researchers comparing these assessment types extensively. For example, Porquet-Lupine and Brigham [2023] explored the combination of peer evaluation with oral assessments to enhance accountability and understanding of team contributions, providing a multifaceted evaluation approach. Candrlic [2020] analysed peer, self, team, and teacher assessments, finding significant differences between team and other grading methods. Interestingly, teacher and peer assessments showed the highest concordance, supporting the reliability of peer evaluation as a substitute or complement to teacher grading, so it is no surprise that the majority of our interviewees employed these approaches in their assessment practice. In the SLR, and anecdotally from some of the interview participants, gender dynamics emerged as a key factor, with team members applying milder criteria for teammates that were men compared to women. These findings underscore the social influences inherent in collaborative assessment contexts. However, Lu and Chiu [2022] highlighted the positive bias of peer assessments (PA) compared to teacher assessments (TA), especially for low-performing students. Teamwork guidelines can mitigate discrepancies, aligning PAs more closely with TAs and enhancing students' perceptions of peer evaluation as a learning tool. The guidelines can amplify the perceived educational value of peer assessment, as students view it as more supportive of learning when such frameworks are in place.

Meanwhile, our findings from the SLR showed researchers developing frameworks for peer evaluation. For example, Fagerholm and Vihavainen [2013] proposed a framework to assess tacit skills through nine indicators for self and peer assessment. Applied across capstone projects, this approach reduced teacher workload while enabling a comprehensive evaluation of student competencies, demonstrating the practicality of structured frameworks in large-scale project settings.

The research evaluated platforms developed for group assessment [Neznanov and Maksimenkova 2016] and forming frameworks to support practitioners applying the group assessment approach in learning environments [Avila et al. 2021], yet there is evidence that students' preferences can evolve over time [Tafliovich et al. 2015], in terms of preferred assessment methods, collaboration tools, and teamwork dynamics. The change in preference could be influenced by their prior learning experiences, or by their increased understanding of the problem domain. Students' evolving preferences should be considered by educators when designing assessment approaches that better align with students' needs to improve their learning outcomes.

6.6 Tools

The process of managing and assessing group projects is complex and a substantial amount of work. In many aspects of computer science education, instructors make use of or create tools to reduce their workload or improve their teaching practices [Blanchard et al. 2022]. Our SLR found 50% of the papers describing novel and existing tools in support of educators, while our interview results showed tools as a central topic for our participants. An important theme about tools supporting group activities concerns the dual role of technology as both a pedagogical tool and a mechanism for workload management. Tools such as Git, Slack, and peer assessment platforms support project work and monitoring, but their effectiveness often depends on instructor familiarity, student digital literacy, and institutional support. Moreover, as cautioned by some of our interviewees, the informal culture associated with some platforms, such as Discord, introduces questions about professionalism, inclusivity, and boundaries in academic contexts.

We found the SPARK tool to be appreciated by students for its perceived fairness in final grades [Kay 2022]. It combines assignment scores with peer assessment participation, where students rate their peers' work on a 0–2 scale. SPARK emphasises participation and fairness while integrating seamlessly into weekly assignment grading. However, the authors recommend SPARK primarily for low-stakes assignments, as it may not be suitable for determining final grades [Kay 2022].

Our results showed innovative tools and methods enriching assessment practices. For example, the mobile app Daily Smirk[20] has demonstrated promise in enhancing peer evaluation experiences [Lu and van der Hoek 2023]. Other approaches utilize objective metrics, such as GitHub contributions, to predict team performance. Analysing metrics like public commits and repository sizes achieved a 70% accuracy in classifying team performance levels [Cui et al. 2022]. Similarly, correlating chat activity on platforms like Slack with GitHub contributions and peer evaluations has revealed significant relationships between communication and performance [Hundhausen et al. 2023]. These findings showed these tools can support educators in evaluating the individual's contributions, and identified a correlation between their contributions to the project and feedback from the activity's peer assessment. These same tools were also utilised by many of our interviewees for assessment. However, during the interviews, some instructors cautioned about the ease with which these tools can be manipulated for the purposes of deception.

Lastly, we found tools and more general applications or software systems used to facilitate group projects. Some of these tools can be used to reduce the workload of the instructor and others will enable more to be achieved with the same amount of resources. Below, we highlight some of the available tools we found and the areas which they can be applied in for student learning (links to most of these can be found in Section 5.6 where tools were first formally discussed).

[20] https://dailysmirk.app/

6.6.1 Communication. Slack, Microsoft Teams, Mattermost, and Discord were all discussed by interviewees as the tool that they chose to enable communication within project groups. The applications differ in terms of price, features available, level of control/moderation possible, as well as student attitudes towards them. Discord was specifically singled out by multiple interviewees as a tool that is commonly used socially by students which in is likely to contribute to unprofessional behavior and language in communications. The SLR papers focused on Google Community Tools and GitHub as available tools educators can adopt for collaboration in their group assessments.

6.6.2 Code Management. Git stands as the de facto standard tool used for version control in software projects, but alternatives such as Mercurial[21] and Perforce[22] may also be useful. Both GitHub and GitLab (including self-hosted) were commonly mentioned in the SLR and interview results as the web application used to manage the repositories of group projects. It was not clear in all cases whether this included the use of features such as issue tracking, pull requests, CI/CD, Github Classroom, or was simply for managing source code. Future research could further examine these features to support various aspects of group assessments.

Additionally, as the de facto standard, many tools have been created which can integrate with or operate on the data generated by Git. The process of managing many repositories for student groups can become complicated, one approach to address this difficulty is RepoBee[23] [Larsén and Glassey 2019], a tool with operations that can manage multiple repositories with a single command. Gource [Caudwell 2010] is another exemplar tool that analyses the logs of various version control systems, such as Git and Mercurial, within a repository to produce animated videos showing the files contributed to or changed by different users over time. This can help students better understand and appreciate collaborative dynamics in group projects.

6.6.3 Coordination/Management. Many existing commercial tools can be used to help manage the completion of group projects within learning environments. While Jira and Trello may be the best known tools for this use case, changes to pricing models or limitations in numbers of users may discourage their use in educational contexts. Alternatives exist such as GitHub Projects or Microsoft Planner that may be free or already included in existing software licenses.

6.6.4 Peer Assessment. One area where tools are frequently used is in the gathering and analysis of peer feedback, which may relate to the higher interest in peer assessment found both in our SLR and interview results. A common approach in both our interviews and SLR was the use of Microsoft/Google Forms to gather the necessary feedback from students. These are not dedicated tools for analysing the feedback and required manual analysis by the educator. A number of alternatives have been developed to manage the gathering and analysis of peer feedback in a more uniform way, such as the previously mentioned mail-based tool PASCA [Neznanov and Maksimenkova 2016], providing a more comprehensive understanding of individual contributions and performance in collaborative projects, thereby addressing common challenges related to assessment fairness and bias. Some tools can be configured to collect Peer Review data, some to collect Peer Evaluation data and some can do both. For example, SPARKPlus [Ahammed 2019], FeedbackFruits [Campbell et al. 2020], CATME [Loughry et al. 2014], and Daily Smirk [Lu and van der Hoek 2023] all provide mechanisms for gathering individual feedback about members of the group project. These tools can provide analysis of the data as well as warnings when some undesirable situations such as conflicts or manipulation.

[21] https://www.mercurial-scm.org/
[22] https://www.perforce.com/
[23] https://repobee.org/

7 Practical Recommendations

From a practical standpoint, educators designing group project assessments should explicitly align learning outcomes, assessment tasks, and evidence-gathering strategies.

Greater transparency in reporting assessment designs—ideally informed by existing and known frameworks—would improve comparability, reproducibility, and knowledge-building in the field. Institutions should also invest in accessible, scalable tools that balance fairness, efficiency, and educational value, particularly in resource-constrained settings.

While different educational contexts can shape the way in which group assessments are performed, there are some general considerations. From those interviewed, some were willing to provide advice regarding group project assessment, based on their own experience, on what to incorporate (do's) and what to avoid (don'ts).

7.1 Exploring *Do's*

- Proactively speak to all teams to suss out potential issues. Be aware of conflicts within teams and address them early, safeguarding groups from manifesting larger more volatile issues. Such conflict within teams can have impacts on student motivation, performance as well as impacting their overall grade for the assessment.
- Students should, where possible, be given thorough feedback regarding their individual grade, with the idea of providing context to the student. Their feedback should explain the final grade they received, the use of a detailed rubric can also provide much needed clarity to students.
- Making use of formative assessment methods, alongside leveraging self-reflective exercises with students can be invaluable in preventing students from inflating their performance.
- Utilize assessments with a variety of different elements, which can allow for a wider group of students to contribute towards.
- Train all personnel involved in the assessment process. Utilize tools that foster professionalism.
- Encourage students to create and agree on a manifesto, team agreement, or code of conduct at the start of their project. This allows students to provide clear expectations to each other regarding their group work going forward. It can also provide students with a clear goal of what is expected of them, within their own group.
- Students concerns should be taken seriously, a student who approaches in confidence should not have issues dismissed as a learning experience without investigation. These concerns could be quite serious issues that require staff member intervention to ameliorate.
- Provide opportunities for students efforts to be acknowledged, for example providing a simple reward for the best presentation. Another example could be sharing a group's achievements on platforms like LinkedIn.

7.2 Exploring *Dont's*

- Peer evaluations should not be used as a punitive tool, This can easily create toxic group dynamics and sow discord. Instead, create clear expectations for students to engage with the process in an equitable manner. Address issues raised by students proactively during the project to foster collaboration and enable the team to course correct.
- When grading teams avoid providing the same grade to all group members, this can foster resentment amongst team members. While it may not be immediately obvious, that resentment can lead to complications in the future. Instead, strive to ensure that the assessment

process properly reflects individual contributions students have made to the project. Keep in mind that the most vocal students are not necessarily contributing more.

- Providing students with a clear rubric and detailed guidelines can be an effective measure to prevent confusion amongst students and also prevent excessive time and energy being spent answering student queries. These resources should be provided to students in a timely fashion and should not change during the assessment process. However, instructors should avoid placing undue importance on number of commits, lines of code, number of comments in communication and other quantitative measures that can be easily manipulated.
- There are conflicting thoughts around the facilitation of team forming, with some advice warding others from controlling the team formation process and some strongly encouraging staff to construct teams for students to prevent future problems developing within the team, especially around gender and cultural imbalances.
- The design of assessment timelines, should take into account deadlines, ensuring they are well-spaced out and manageable for the students to accomplish. This can support students well-being and minimize student stress.
- Interviewed participants stated that students generally appreciate clarity regarding assessment criteria - either through a clear grading rubric, shared in advance, or through the use of exemplar materials showcasing both good and bad works. Notably some interviewees highlighted that students, in some cultural contexts, did not always feel comfortable passing judgment on their peers during a peer review process. The students did however value having their voices heard. One point of student feedback that was raised, was regarding fairness, as outlined in section 5.7.4. One instructor theorized that students "more than anything, hate carrying others". Another mentioned receiving negative input about sharing grades "so and so didn't pull their weight and we all get the same grade".

8 Limitations

This study has limitations. Firstly, the group confined the literature search to papers written in English in well-known databases, which may have excluded valuable perspectives and research published in other languages or from regional conferences. It may skew the findings, particularly at institutions that do not offer courses in English. Furthermore, the group excluded short publications under four pages due to time and resource constraints. As such, our analysis has not incorporated potentially valuable insights from shorter pieces.

Methodologically, this review encountered challenges common to synthesising educational assessment literature, including inconsistent reporting, varied study quality, and a scarcity of robust evaluations. This includes varying terminology researchers use to describe studies that focus on group assessment. As a result, our SLR may have missed relevant literature.

In terms of representation, notable geographical gaps remain. Few studies originated from Eastern, South-Eastern, or Central Asia, Latin America, Northern Africa, and none were found from Sub-Saharan Africa. This disparity mirrors broader patterns in computing education research [Blanchard et al. 2022] and highlights the need for more diverse perspectives. This absence limits understanding of assessment practices in regions where cultural, institutional, and educational contexts may differ substantially from those in the predominantly represented areas. Addressing these omissions is essential for developing a truly global perspective on group project assessment in computing education.

The semi-structured protocol for the interviews might have introduced some bias. The approach helped facilitate a focused discussion, however, it also directed the conversation toward particular themes. For example, interviewers asked about course context, assessment structures, tools, and the balance between individual and collective measures. The broad questions posed toward the end of the interview about challenges and opportunities for improvement in group project assessment also directed interviewees toward discussing issues they felt were most relevant or pressing within their contexts. While these factors were necessary to guide the interviews, they may have introduced a bias in the content of the data collected in the process.

Of the 18 interviews conducted, eight were with co-authors of this study. Each member met the inclusion criteria. Three members of the team built the interview protocol and two of them were interviewed for the study, so they knew the questions ahead of time. The remaining 6 internal interviewees heard the exact questions for the first time during their interviews. However, their knowledge of the subject matter and their role within the group may have shaped their responses.

The interview research sub-team strived to gather a global sample. Unfortunately, there are notable geographical gaps. Specifically, the we did not interview instructors from Sub-Saharan African or South American institutions, potentially reducing the perspectives captured. This absence could have resulted in a limited understanding of assessment practices in regions where contexts may differ significantly from those represented in the sample.

Because the research team conducted the content analysis and interviews in parallel, one did not influence the other. Therefore, the convergence of the two datasets as the data approaches saturation is reassuring. However, it may also suggest confirmation bias in the analysis process.

9 Conclusion

This article examines group projects in undergraduate computing education, identifying current trends in project contexts, assessment approaches, and tool adoption. Group projects remain concentrated in software engineering courses and capstone experiences, with fewer instances in specialised or interdisciplinary contexts. Assessment methods are diverse, with presentations, reports, and inspections of technical artefacts commonly employed. We present a schema of 23 approaches drawn from the literature. Peer evaluation and analytics derived from version control systems are increasingly used to track individual contributions and provide formative feedback. However, evidence of the effectiveness of these practices remains limited and inconsistent.

We found that there is no one-size-fits-all method of assessment suited to handle the variety of contexts in which educators apply group projects for student learning. Instead, our work offers approaches to assessments that educators can evaluate and adopt, with learning outcomes similar to those in their courses. This article provides educators with the opportunity to explore various assessment approaches, detailed in section 4, that others have successfully implemented with similar goals and learning outcomes. This work also lists some practical recommendations to assessment of group work in section 7.

A recurring concern in the literature and among practitioners is fairness in assessment, particularly in addressing issues such as unequal participation and free-riding. While various strategies—contribution analytics, structured observations, and especially peer and self evaluations—promise to mitigate these challenges, there is little consensus on their validity, reliability, or the extent of the mitigation. Moreover, a common limitation in published studies is the lack of comprehensive descriptions of assessment designs, with many addressing only isolated components of assessment frameworks, such as Almond's Model. Greater transparency and standardisation in reporting assessment practices would support more meaningful comparisons and evidence synthesis.

Finally, while educators acknowledge issues of equity, justice, and inclusivity in assessment practices as significant challenges, empirical studies examining these dimensions remain scarce. Future research should prioritise understanding the cultural and structural factors that shape group project experiences and assessment outcomes globally. Addressing these gaps is essential for developing fair, effective, and contextually appropriate assessment strategies that support students' professional development and employability. Future research could address the identified gaps, particularly in terms of assessment effectiveness, the equitable use of technology, and cross-cultural differences in group project assessment. Expanding the evidence base will be essential for developing fair, pedagogically sound, and globally relevant assessment practices in computing education. Expanding the evidence base, particularly regarding the effectiveness of assessments, the equitable use of technology, and cross-cultural differences in group project assessments, will be essential.

References

Faisal Ahammed. 2019. Peer Assessment Using SPARKPLUS for Engineering Students. In *2019 International Joint Conference on Information, Media and Engineering (IJCIME)*. IEEE, New York, 112–115. doi:10.1109/IJCIME49369.2019.00031

Efthimia Aivaloglou and Anna van der Meulen. 2021. An Empirical Study of Students' Perceptions on the Setup and Grading of Group Programming Assignments. *ACM Trans. Comput. Educ.* 21, 3, Article 17 (mar 2021), 22 pages. doi:10.1145/3440994

Jordan Allison, Abu Alam, Luke Gassmann, Gareth Nelson, and Kamal Zidan and. 2024. Fostering the development of computer science graduate employability through agile projects. *Journal of Further and Higher Education* 48, 4 (2024), 417–435. doi:10.1080/0309877X.2024.2340642

Russell Almond, Linda Steinberg, and Robert Mislevy. 2002. Enhancing the design and delivery of assessment systems: A four-process architecture. *The Journal of Technology, Learning and Assessment* 1, 5 (2002), 64 pages. https://ejournals.bc.edu/index.php/jtla/article/view/1671

William F. Atchison, Samuel D. Conte, John W. Hamblen, Thomas E. Hull, Thomas A. Keenan, William B. Kehl, Edward J. McCluskey, Silvio O. Navarro, Werner C. Rheinboldt, Earl J. Schweppe, William Viavant, and David M. Young. 1968. Curriculum 68: Recommendations for academic programs in computer science: a report of the ACM curriculum committee on computer science. *Commun. ACM* 11, 3 (March 1968), 151–197. doi:10.1145/362929.362976

Daymy Tamayo Avila, Wim Van Petegem, and Arno Libotton. 2021. ASEST framework: a proposal for improving teamwork by making cohesive software engineering student teams. *European Journal of Engineering Education* 46, 5 (2021), 750–764. doi:10.1080/03043797.2020.1863339

Rosalina Babo, Jarkko T Suhonen, and Markku Tukiainen. 2020. Improving workgroup assessment with WebAVALIA: The concept, framework and first results. *Journal of Information Technology Education. Innovations in Practice* 19 (2020), 157.

Adelina Basholli, Fesal Baxhaku, Dimitris Dranidis, and Thanos Hatziapostolou. 2013. Fair assessment in software engineering capstone projects. In *Proceedings of the 6th Balkan Conference in Informatics* (Thessaloniki, Greece) *(BCI '13)*. Association for Computing Machinery, New York, NY, USA, 244–250. doi:10.1145/2490257.2490268

Maria Cecilia Bastarrica, Daniel Perovich, and Maira Marques Samary. 2017. What Can Students Get from a Software Engineering Capstone Course?. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)* (Buenos Aires, Argentina) *(ICSE-SEET '17)*. IEEE, New York, 137–145. doi:10.1109/icse-seet.2017.15

Margaret Bearman, Phillip Dawson, David Boud, Sue Bennett, Matt Hall, and Elizabeth Molloy. 2016. Support for assessment practice: developing the Assessment Design Decisions Framework. *Teaching in Higher Education* 21, 5 (2016), 545–556. doi:10.1080/13562517.2016.1160217

R Meredith Belbin. 2010. *Management Teams*. Routledge, London. doi:10.4324/9780080963594

Mario Bernhart, Thomas Grechenig, Jennifer Hetzl, and Wolfgang Zuser. 2006. Dimensions of software engineering course design. In *Proceedings of the 28th international Conference on Software Engineering*. 667–672.

Shreeranga Bhat, Sathyendra Bhat, Ragesh Raju, Rio D'Souza, and Binu K.G. 2020. Collaborative Learning for Outcome Based Engineering Education: A Lean Thinking Approach. *Procedia Computer Science* 172 (2020), 927–936. doi:10.1016/j.procs.2020.05.134 9th World Engineering Education Forum (WEEF 2019) Proceedings : Disruptive Engineering Education for Sustainable Development.

John Biggs. 1996. Enhancing teaching through constructive alignment. *Higher education* 32, 3 (1996), 347–364. doi:10.1007/BF00138871

John Biggs. 2003. *Teaching for Quality Learning at University: What the Student Does*. McGraw-Hill Education, Columbus, OH. https://www.researchgate.net/publication/215915395_Teaching_for_Quality_Learning_at_University

S. E. Binns and A. R. West. 1975. *An Undergraduate Group Project in Software Engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg, 679–685. doi:10.1007/978-3-662-40087-6_63

Brian M Blake. 2005. Integrating large-scale group projects and software engineering approaches for early computer science courses. *IEEE Transactions on Education* 48, 1 (2005), 63–72.

Jeremiah Blanchard, John R. Hott, Vincent Berry, Rebecca Carroll, Bob Edmison, Richard Glassey, Oscar Karnalim, Brian Plancher, and Seán Russell. 2022. Stop Reinventing the Wheel! Promoting Community Software in Computing Education. In *Proceedings of the 2022 Working Group Reports on Innovation and Technology in Computer Science Education* (Dublin, Ireland) *(ITiCSE-WGR '22)*. Association for Computing Machinery, New York, NY, USA, 261–292. doi:10.1145/3571785.

3574129

Grant Braught, John Maccormick, James Bowring, Quinn Burke, Barbara Cutler, David Goldschmidt, Mukkai Krishnamoorthy, Wesley Turner, Steven Huss-Lederman, Bonnie Mackellar, and Allen Tucker. 2018. A Multi-Institutional Perspective on H/FOSS Projects in the Computing Curriculum. *ACM Trans. Comput. Educ.* 18, 2, Article 7 (July 2018), 31 pages. doi:10.1145/3145476

Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2 (2006), 77–101. doi:10.1191/1478088706QP063OA

Sally Brown and Peter Knight. 1994. *Assessing Learners in Higher Education* (1st ed.). Routledge, Milton Park, UK. doi:10.4324/9780203062036

Bernd Bruegge, Stephan Krusche, and Lukas Alperowitz. 2015. Software engineering project courses with industrial clients. *ACM Transactions on Computing Education (TOCE)* 15, 4 (2015), 1–31.

Kevin Buffardi. 2020. Assessing Individual Contributions to Software Engineering Projects with Git Logs and User Stories. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education* (Portland, OR, USA) *(SIGCSE '20)*. Association for Computing Machinery, New York, NY, USA, 650–656. doi:10.1145/3328778.3366948

Alison Burke. 2011. Group work: How to use groups effectively. *Journal of Effective Teaching* 11, 2 (2011), 87–95.

Chris Campbell, Lenka Borer, and Sheila McCarthy. 2020. Enhancing feedback with FeedbackFruits: Scaling for transformative implementation. In *ASCILITE's First virtual conference*, Sue Gregory, Steve Warburton, and Mitchell Parkes (Eds.). University of New England, Australia, 322–327. https://researchoutput.csu.edu.au/en/publications/enhancing-feedback-with-feedbackfruits-scaling-for-transformative/ Includes bibliographical references.; 37th International Conference of Innovation, Practice and Research in the Use of Educational Technologies in Tertiary Education : ASCILITE 2020 ; Conference date: 30-11-2020 Through 01-12-2020.

Sanja Candrlic. 2020. Different Perspectives on Students' Presentation Skills: Teacher, Peer, Team And Self-Assessment. In *2020 15th International Conference on Computer Science & Education (ICCSE)*. IEEE, New York, 156–161. doi:10.1109/ICCSE49874.2020.9201796

M Ariel Cascio, Eunlye Lee, Nicole Vaudrin, and Darcy A Freedman. 2019. A team-based approach to open coding: Considerations for creating intercoder consensus. *Field methods* 31, 2 (2019), 116–130. doi:10.1177/1525822X19838237

Andrew H. Caudwell. 2010. Gource: visualizing software version control history. In *Proceedings of the ACM International Conference Companion on Object Oriented Programming Systems Languages and Applications Companion* (Reno/Tahoe, Nevada, USA) *(OOPSLA '10)*. Association for Computing Machinery, New York, NY, USA, 73–74. doi:10.1145/1869542.1869554

María Cecilia Bastarrica, Francisco J. Gutierrez, María Marques, and Daniel Perovich. 2023. On the Impact of Grading on Teamwork Quality in a Software Engineering Capstone Course. *IEEE Access* 11 (2023), 36492–36503. doi:10.1109/ACCESS.2023.3265302

Robert Chatley and Tony Field. 2017. Lean Learning - Applying Lean Techniques to Improve Software Engineering Education. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET)*. IEEE, New York, 117–126. doi:10.1109/ICSE-SEET.2017.5

Hsi-Min Chen, Bao-An Nguyen, and Chyi-Ren Dow. 2022. Code-quality evaluation scheme for assessment of student contributions to programming projects. *Journal of Systems and Software* 188 (2022), 111273. doi:10.1016/j.jss.2022.111273

Eva Hammar Chiriac. 2014. Group work as an incentive for learning–students' experiences of group work. *Frontiers in psychology* 5 (2014), 558.

Vanessa Maribel Choque-Soto and Victor Dario Sosa-Jauregui. 2024. Assessing Soft Skills Development in Informatics Students Through Project-Based Learning and Agile Frameworks. In *2024 International Symposium on Accreditation of Engineering and Computing Education (ICACIT)*. IEEE, New York, 1–8. doi:10.1109/ICACIT62963.2024.10788614

Stefan C. Christov and Mark E. Hoffman. 2019. Experiential Learning of Software Project Management and Software Development via Course Collaboration. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education* (Minneapolis, MN, USA) *(SIGCSE '19)*. Association for Computing Machinery, New York, NY, USA, 160–166. doi:10.1145/3287324.3287457

Alison Clear, Allen Parrish, John Impagliazzo, Pearl Wang, Paolo Ciancarini, Ernesto Cuadros-Vargas, Stephen Frezza, Judith Gal-Ezer, Arnold Pears, Shingo Takada, Heikki Topi, Gerrit van der Veer, Abhijat Vichare, Les Waguespack, and Ming Zhang. 2020. *Computing Curricula 2020.* ACM, New York. https://users.cs.fiu.edu/~prabakar/ugc/ACM_Reports/2020_ACM-IEEE_CC_Draft.pdf

Tony Clear. 2009. Thinking ISsues: the three p's of capstone project performance. *ACM SIGCSE Bulletin* 41, 2 (2009), 69–70. https://www.researchgate.net/publication/220612736_Thinking_ISsues_the_three_p's_of_capstone_project_performance

Elizabeth G Cohen. 1994. Restructuring the classroom: Conditions for productive small groups. *Review of educational research* 64, 1 (1994), 1–35.

Tom Crick, Tom Prickett, Jill Bradnum, and Alan Godfrey. 2022. Gender parity in peer assessment of team software development projects. In *Proceedings of the 6th Conference on Computing Education Practice* (Durham, United Kingdom) *(CEP '22)*. Association for Computing Machinery, New York, NY, USA, 9–12. doi:10.1145/3498343.3498346

Jialin Cui, Ruochi Li, Kaida Lou, Chengyuan Liu, Yunkai Xiao, Qinjin Jia, Edward Gehringer, and Runqiu Zhang. 2022. Can pre-class github contributions predict success by student teams?. In *Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET '22)*. Association for Computing Machinery, New York, NY, USA, 40–49. doi:10.1145/3510456.3514144

W Martin Davies. 2009. Groupwork as a form of assessment: Common problems and recommended solutions. *Higher education* 58, 4 (2009), 563–584.

Barbara Gross Davis. 1993. *Tools for teaching*. John Wiley & Sons, Hoboken, NJ.

L. Dorić, N. Luburić, J. Slivka, and A. Kovačević. 2023. Understanding the Teamwork Challenges of Software Engineering Students. In *2023 46th MIPRO ICT and Electronics Convention (MIPRO)*. IEEE, New York, 1578–1583. doi:10.23919/MIPRO57284.2023.10159643

Robert F Dugan Jr. 2011. A survey of computer science capstone course literature. *Computer Science Education* 21, 3 (2011), 201–267.

Rosanne English and Alan Hayes. 2022. Towards integrated graduate skills for UK computing science students. In *Proceedings of the 2022 Conference on United Kingdom & Ireland Computing Education Research*. 1–7.

Fabian Fagerholm and Arto Vihavainen. 2013. Peer assessment in experiential learning Assessing tacit and explicit skills in agile software engineering capstone projects. In *2013 IEEE Frontiers in Education Conference (FIE)*. IEEE, New York, 1723–1729. doi:10.1109/FIE.2013.6685132

Nancy Falchikov. 1995. Peer feedback marking: Developing peer assessment. *Innovations in Education and training International* 32, 2 (1995), 175–187. doi:10.1080/1355800950320212

Sally Fincher, Marian Petre, and Martyn Clark. 2001. *Computer science project work: principles and pragmatics*. Springer Science & Business Media, Germany.

L Dee Fink. 2013. *Creating significant learning experiences: An integrated approach to designing college courses*. John Wiley & Sons, Hoboken, NJ.

Matthew Forshaw, Cristina Adriana Alexadru, Caitlin Bentley, Vladimiro González-Zelaya, Joseph Kwame Adjei, Vangel Ajanovski, Mireilla Bikanga Ada, Julian Brooks, Joshua Burridge, Alex Chao, Rutwa Engineer, Olga Glebova, Tasmina Islam, Mitsuka Kiyohara, Shao-Heng Ko, Ellert Smári Kristbergsson, Svetlana Peltsverger, Seán Russell, Maíra Marques Samary, Merel Steenbergen, and Carolin Wortmann. 2025. Fairness in Student Allocation and Group Formation. In *Proceedings of the 30th ACM Conference on Innovation and Technology in Computer Science Education V. 2* (Nijmegen, Netherlands) *(ITiCSE 2025)*. Association for Computing Machinery, New York, NY, USA, 699–700. doi:10.1145/3724389.3731280

Vahid Garousi, Görkem Giray, Eray Tüzün, Cagatay Catal, and Michael Felderer. 2019. Aligning software engineering education with industrial needs: A meta-analysis. *Journal of Systems and Software* 156 (2019), 65–83. doi:10.1016/j.jss.2019.06.044

Nasser Giacaman and Oliver Sinnen. 2018. Preparing the software engineer for a modern multi-core world. *J. Parallel and Distrib. Comput.* 118 (2018), 247–263. doi:10.1016/j.jpdc.2018.02.028

Graham Gibbs. 1988. *Learning by doing: A guide to teaching and learning methods*. Further Education Unit, Oxford Polytechnic, Oxford, UK. https://thoughtsmostlyaboutlearning.wordpress.com/wp-content/uploads/2015/12/learning-by-doing-graham-gibbs.pdf

Neil Andrew Gordon. 2010. Group working and peer assessment—using WebPA to encourage student engagement and participation. *Innovation in Teaching and Learning in Information and Computer Sciences* 9, 1 (2010), 20–31. doi:10.11120/ital.2010.09010020

Wouter Groeneveld, joost Vennekens, and kris Aerts. 2019. Software Engineering Education Beyond the Technical: A Systematic Literature Review. In *Proceedings of the 47th SEFI Conference 2019*, Vol. 47. SEFI - European Society for Engineering Education; Brussels; https://www.sefi.be/proceedings/, Brussels, Belgium, 1607–1622. doi:10.48550/arXiv.1910.09865

Fabian Gunnars and Adam Palmquist. 2026. Teaching game development at universities: A systematic literature review of educational practices for multidisciplinary industrial demands. *Games and Culture* (2026), 15554120261442965.

Michael Guttmann, Aleksandar Karaka, and Denis Helic. 2024. Attribution of Work in Programming Teams with Git Reporter. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (Portland, OR, USA) *(SIGCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 436–442. doi:10.1145/3626252.3630785

Stephanie J. Hanrahan and Geoff Isaacs. 2001. Assessing Self- and Peer-assessment: The students' views. *Higher Education Research & Development* 20, 1 (2001), 53–70. doi:10.1080/07294360123776

Robert Hans. 2024. Systematic review of software project success criteria from future software practitioners' perspective. *Procedia Computer Science* 239 (2024), 1289–1297. doi:10.1016/j.procs.2024.06.298 CENTERIS – International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies 2023.

Alison Harvey. 2019. Becoming Gamesworkers: Diversity, Higher Education, and the Future of the Game Industry. *Television & New Media* 20, 8 (2019), 756–766. doi:10.1177/1527476419851080

Andrew F Hayes and Klaus Krippendorff. 2007. Answering the call for a standard reliability measure for coding data. *Communication methods and measures* 1, 1 (2007), 77–89. doi:10.1080/19312450709336664

Nicole Herbert. 2007. Quantitative Peer Assessment: Can Students Be Objective?. In *Proceedings of the Ninth Australasian Conference on Computing Education - Volume 66* (Ballarat, Victoria, Australia) *(ACE '07)*. Australian Computer Society, Inc., AUS, 63–71. https://dl.acm.org/doi/abs/10.5555/1273672.1273680

Nicole Herbert. 2018. Reflections on 17 years of ICT Capstone Project Coordination: Effective Strategies for Managing Clients, Teams and Assessment. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (Baltimore, Maryland, USA) *(SIGCSE '18)*. Association for Computing Machinery, New York, NY, USA, 215–220. doi:10.1145/3159450.3159584

Haneen Hijazi and Yasser Ali Alshehri. 2024. Leveraging Peer-Assessment in Project-Based Software Engineering Courses. In *2024 36th International Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, New York, 1–7. doi:10.1109/CSEET62301.2024.10663031 ISSN: 2377-570X.

Christopher K. Hobbs and Herbert H. Tsang. 2014. Industry in the Classroom: Equipping Students with Real-World Experience A reflection on the effects of industry partnered projects on computing education. In *Proceedings of the Western Canadian Conference on Computing Education* (Richmond, BC, Canada) *(WCCCE '14)*. Association for Computing Machinery, New York, NY, USA, Article 7, 5 pages. doi:10.1145/2597959.2597967

Quan Nha Hong, Sergi Fàbregues, Gillian Bartlett, Felicity Boardman, Margaret Cargo, Pierre Dagenais, Marie-Pierre Gagnon, Frances Griffiths, Belinda Nicolau, Alicia O'Cathain, et al. 2018. The Mixed Methods Appraisal Tool (MMAT) version 2018 for information professionals and researchers. *Education for information* 34, 4 (2018), 285–291. https://journals.sagepub.com/doi/abs/10.3233/EFI-180221

Sara Hooshangi, Asma Shakil, Subhasish Dasgupta, Karen C. C. Davis, Mohammed Farghally, KellyAnn Fitzpatrick, Mirela Gutica, Ryan Hardt, Steve Riddle, and Mohammed Seyam. 2025. Instructors' Perspectives on Capstone Courses in Computing Fields: A Mixed-Methods Study. In *2024 Working Group Reports on Innovation and Technology in Computer Science Education* (Milan, Italy) *(ITiCSE 2024)*. Association for Computing Machinery, New York, NY, USA, 68–94. doi:10.1145/3689187.3709608

Sara Hooshangi, Asma Shakil, Steve Riddle, Ilknur Aydin, Nayla Nasir, Tejasvi Parupudi, Attiqa Rehman, Michael James Scott, Jan Vahrenhold, Amali Weerasinghe, and Xi Wu. 2026. Evaluating Assessment Practices in Team-Based Computing Capstone Projects. In *Proceedings of the 2025 Working Group Reports on Innovation and Technology in Computer Science Education* (Netherlands) *(ITiCSE-WGR 2025)*. Association for Computing Machinery, New York, NY, USA, 277–312. doi:10.1145/3760545.3783974

Christopher Hundhausen, Phill Conrad, Olusola Adesope, and Ahsun Tariq. 2023. Combining GitHub, Chat, and Peer Evaluation Data to Assess Individual Contributions to Team Software Development Projects. *ACM Trans. Comput. Educ.* 23, 3 (July 2023), 33:1–33:23. doi:10.1145/3593592

Ville Isomöttönen. 2014. Making group processes explicit to student: a case of justice. In *Proceedings of the 2014 Conference on Innovation & Technology in Computer Science Education* (Uppsala, Sweden) *(ITiCSE '14)*. Association for Computing Machinery, New York, NY, USA, 195–200. doi:10.1145/2591708.2591717

Ville Isomöttönen and Emmi Ritvos. 2021. Digging into group establishment: Intervention design and evaluation. *Journal of Systems and Software* 178 (2021), 110974. doi:10.1016/j.jss.2021.110974

Tineke Jacobs. 2024. Grading individual students in software engineering projects confidently. In *Proceedings of the 29th European Conference on Pattern Languages of Programs, People, and Practices (EuroPLoP '24)*. Association for Computing Machinery, New York, NY, USA, Article 13, 10 pages. doi:10.1145/3698322.3698330

David W Johnson and Roger T Johnson. 1999. Making cooperative learning work. *Theory into practice* 38, 2 (1999), 67–73.

M. Kaluza, S. Candrlic, and M. Asenbrener Katic. 2023. Hybrid Agile Approach in Software Engineering Education – A Case Study. In *2023 46th MIPRO ICT and Electronics Convention (MIPRO)*. IEEE, New York, 1543–1548. doi:10.23919/MIPRO57284.2023.10159921 ISSN: 2623-8764.

Sam Kavanagh and Andrew Luxton-Reilly. 2016. Rubrics used in peer assessment. In *Proceedings of the Australasian Computer Science Week Multiconference* (Canberra, Australia) *(ACSW '16)*. Association for Computing Machinery, New York, NY, USA, Article 10, 6 pages. doi:10.1145/2843043.2843347

Jennifer S. Kay. 2022. Peer Grading Without Protest: The SPARK Approach to Summative Peer Assessment. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1* (Providence, RI, USA) *(SIGCSE 2022)*. Association for Computing Machinery, New York, NY, USA, 119–125. doi:10.1145/3478431.3499284

Harold Kerzner. 2017. *Project Management: A Systems Approach to Planning, Scheduling, and Controlling* (12th ed.). John Wiley & Sons, Hoboken, NJ. https://www.wiley.com/en-se/Project+Management%3A+A+Systems+Approach+to+Planning%2C+Scheduling%2C+and+Controlling%2C+12th+Edition-p-9781119165361

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

Diana Kirk, Andrew Luxton-Reilly, and Ewan Tempero. 2022. Refining a Risk Framework for Student Group Projects. In *Proceedings of the 22nd Koli Calling International Conference on Computing Education Research* (Koli, Finland) *(Koli Calling '22)*. Association for Computing Machinery, New York, NY, USA, Article 19, 11 pages. doi:10.1145/3564721.3564730

Barbara Kitchenham. 2004. Procedures for performing systematic reviews. *Keele, UK, Keele University* 33, 2004 (2004), 1–26. https://www.inf.ufsc.br/~aldo.vw/kitchenham.pdf

Mikko Kokkoniemi and Ville Isomöttönen. 2023. A systematic mapping study on group work research in computing education projects. *Journal of Systems and Software* 204 (2023), 111795. doi:10.1016/j.jss.2023.111795

David R Krathwohl. 2002. A revision of Bloom's taxonomy: An overview. *Theory into practice* 41, 4 (2002), 212–218.

Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology.* Sage publications, London. https://web.stanford.edu/class/comm1a/readings/krippendorf-ch1and4.pdf

Vasilka Krumova, Dancho Danalev, and Senya Terzieva. 2020. Project based learning-strategy for professional competence development in higher education. *Science, Engineering and Education* 5, 1 (2020), 27–40. doi:10.59957/see.v5.i1.2020.6

Emily Rose Lai. 2011. *Collaboration: A Literature Review. Research Report.* Pearson, London, England. https://scispace.com/pdf/collaboration-a-literature-review-1i8atuar0d.pdf

Simon Larsén and Richard Glassey. 2019. RepoBee: Developing Tool Support for Courses using Git/GitHub. In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education* (Aberdeen, Scotland Uk) *(ITiCSE '19)*. Association for Computing Machinery, New York, NY, USA, 534–540. doi:10.1145/3304221.3319784

Patricia Lasserre and Carolyn Szostak. 2011. Effects of team-based learning on a CS1 course. In *Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education* (Darmstadt, Germany) *(ITiCSE '11)*. Association for Computing Machinery, New York, NY, USA, 133–137. doi:10.1145/1999747.1999787

Sherlock A Licorish, Matthias Galster, Georgia M Kapitsaki, and Amjed Tahir. 2022. Understanding students' software development projects: Effort, performance, satisfaction, skills and their relation to the adequacy of outcomes developed. *Journal of Systems and Software* 186 (2022), 111156. doi:10.1016/j.jss.2021.111156

Robert Lingard and Shan Barkataki. 2011. Teaching teamwork in engineering and computer science. In *2011 Frontiers in education conference (FIE)*. IEEE, New York, F1C–1. doi:10.1109/FIE.2011.6143000

Xiongyi Liu and Lan Li. 2014. Assessment training effects on student assessment skills and task performance in a technology-facilitated peer assessment. *Assessment & Evaluation in Higher Education* 39, 3 (2014), 275–292. doi:10.1080/02602938.2013.823540

Angeles Lopez, M. Asunción Castaño, M. Victoria Ibáñez, Ismael Sanz, Lledó Museros, and Reyes Grangel. 2025. Improving the Assessment of Capstone Projects in the Bachelor's Degree in Computer Engineering. *ACM Trans. Comput. Educ.* 25, 2, Article 14 (May 2025), 28 pages. doi:10.1145/3722230

Misty L Loughry, Matthew W Ohland, and David J Woehr. 2014. Assessing teamwork skills for assurance of learning using CATME team tools. *Journal of Marketing Education* 36, 1 (2014), 5–19. https://info.catme.org/wp-content/uploads/Assurance-of-Learning-PDF-2013.08.13.pdf

Meixiu Lu and Ming Ming Chiu. 2022. Do Teamwork Guidelines Improve Peer Assessment Accuracy or Attitudes During Collaborative Learning? *IEEE Transactions on Education* 65, 4 (2022), 485–492. doi:10.1109/TE.2021.3130242

Yu Lu and André van der Hoek. 2023. The Daily Smirk: A Preliminary Prototype for Continuous Peer Assessment of Team-based Projects. In *2023 IEEE 35th International Conference on Software Engineering Education and Training (CSEE&T)* (2023-08). IEEE, New York, 31–40. doi:10.1109/CSEET58097.2023.00014 ISSN: 2377-570X.

Andrew Luxton-Reilly. 2009. A systematic review of tools that support peer assessment. *Computer Science Education* 19, 4 (2009), 209–232. doi:10.1080/08993400903384844

Bonnie K. MacKellar. 2013. Analyzing coordination among students in a software engineering project course. In *2013 26th International Conference on Software Engineering Education and Training (CSEE&T)* (2013-05). IEEE, New York, 279–283. doi:10.1109/CSEET.2013.6595261

Zaigham Mahmood. 2007. A framework for software engineering education: a group projects approach. *International Journal of Education and Information Technologies* 1, 3 (2007), 153–156. https://www.naun.org/main/NAUN/educationinformation/eit-24.pdf

Sharon Mason. 2020. Collaborative Learning in Computing Education: Faculty Perspectives and Practices. In *Proceedings of the 2020 ACM Conference on International Computing Education Research* (Virtual Event, New Zealand) *(ICER '20)*. Association for Computing Machinery, New York, NY, USA, 136–146. doi:10.1145/3372782.3406254

Philipp Mayring. 2014. *Qualitative content analysis - theoretical foundation, basic procedures and software solution.* Beltz, Weinheim. https://www.researchgate.net/publication/266859800_Qualitative_content_analysis_-_theoretical_foundation_basic_procedures_and_software_solution

Jade McKay and Bhavani Sridharan. 2024. Student perceptions of collaborative group work (CGW) in higher education. *Studies in Higher Education* 49, 2 (2024), 221–234. doi:10.1080/03075079.2023.2227677

Matthew B. Miles, A. Michael Huberman, and Johnny Saldaña. 2014. *Qualitative data analysis a methods sourcebook.* SAGE Publications, Inc., Thousand Oaks, Califorinia. https://www.metodos.work/wp-content/uploads/2024/01/Qualitative-Data-Analysis.pdf

Robert J Mislevy, Russell G Almond, and Janice F Lukas. 2003a. *A brief introduction to evidence-centered design.* Technical Report. National Center for Research on Evaluation, Standards, and Student Testin. i–29 pages. https://files.eric.ed.gov/fulltext/ED483399.pdf

Robert J Mislevy and Michelle M Riconscente. 2006. Evidence-centered assessment design. In *Handbook of test development.* Routledge, California, 75–104. doi:10.4324/9780203874776-9

Robert J Mislevy, Linda S Steinberg, and Russell G Almond. 2003b. On the structure of educational assessments. *Measurement: Interdisciplinary research and perspectives* 1, 1 (2003), 3–62. doi:10.1207/S15366359MEA0101_02

Alexander Mitchell, Terry Greer, Warwick New, Joseph Walton-Rivers, Matt Watkins, Douglas Brown, and Michael James Scott. 2021. Student Perspectives on the Purpose of Peer Evaluation During Group Game Development Projects. In *United Kingdom and Ireland Computing Education Research Conference.* (Glasgow, United Kingdom) *(UKICER '21).* Association for Computing Machinery, New York, NY, USA, Article 7, 7 pages. doi:10.1145/3481282.3481294

Alexander Mitchell, Michael Scott, Joseph Walton-Rivers, Matt Watkins, Warwick New, and Douglas Brown. 2022. An exploratory analysis of student experiences with peer evaluation in group game development projects. In *Proceedings of the 2022 Conference on United Kingdom & Ireland Computing Education Research.* 1–7.

Alexander Mitchell, Michael James Scott, Bedour Alshaigy, Wendell Bento Geraldes, Rita Garcia, Mirela Gutica, Hieke Keuning, Ellie Lovellette, P D Parthasarathy, Seán Russell, Sandra Schulz, and Xi Wu. 2024. Exploring Approaches to Assessing Student Teamwork in Undergraduate Computing Projects. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 2* (Virtual Event, NC, USA) *(SIGCSE Virtual 2024).* Association for Computing Machinery, New York, NY, USA, 285–286. doi:10.1145/3649409.3691072

Higinio Mora, María Teresa Signes-Pont, Andrés Fuster-Guilló, and María L Pertegal-Felices. 2020. A collaborative working model for enhancing the learning process of science & engineering students. *Computers in Human Behavior* 103 (2020), 140–150. doi:10.1016/j.chb.2019.09.008

Colin J. Neill, Joanna F. DeFranco, and Raghvinder S. Sangwan. 2017. Improving collaborative learning in online software engineering education. *European Journal of Engineering Education* 42, 6 (2017), 591–602. doi:10.1080/03043797.2016.1203293

Andres Neyem, Jose I. Benedetto, and Andres F. Chacon. 2014. Improving software engineering education through an empirical approach: lessons learned from capstone teaching experiences. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (Atlanta, Georgia, USA) *(SIGCSE '14).* Association for Computing Machinery, New York, NY, USA, 391–396. doi:10.1145/2538862.2538920

Alexey Neznanov and Olga Maksimenkova. 2016. The PASCA: A Mail Based Randomized Blinded Peer Assessment System for Complex Artifacts. *Procedia Computer Science* 96 (2016), 826–837. doi:10.1016/j.procs.2016.08.261 Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 20th International Conference KES-2016.

David J Nicol and Debra Macfarlane-Dick. 2006. Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. *Studies in higher education* 31, 2 (2006), 199–218. https://www.psy.gla.ac.uk/~steve/rap/docs/nicol.dmd.pdf

Emmanuella Onyinye Nwulu, Tari Yvonne Elete, Kingsley Onyedikachi Omomo, OA Akano, and OV Erhueh. 2023. The importance of interdisciplinary collaboration for successful engineering project completions: A strategic framework. *World Journal of Engineering and Technology Research* 2, 3 (2023), 48–56. doi:10.53346/wjetr.2023.2.3.0048

Hege Annette Olstad, Birgit Rognebakke Krogstie, Xiaomeng Su, and Rune Hjelsvold. 2024. Enabling computing students to recognize acquired competencies with ePortfolios. *Computer Science Education* 0, 0 (2024), 1–28. doi:10.1080/08993408.2024.2353482

Matthew J Page, Joanne E McKenzie, Patrick M Bossuyt, Isabelle Boutron, Tammy C Hoffmann, Cynthia D Mulrow, Larissa Shamseer, Jennifer M Tetzlaff, Elie A Akl, Sue E Brennan, et al. 2021. The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *British Medical Journal* 372 (2021), n71. doi:10.1136/bmj.n71

Maximiliano Paredes-Velasco, Mónica Arnal-Palacián, Jaime Urquiza-Fuentes, and Mercedes Martín-Lope. 2023. Improving soft skills through an interdisciplinary approach in a realistic context between education and CS students in an HCI course. *IEEE Transactions on Education* 66, 6 (2023), 579–590.

Karen L Peel. 2020. A beginner's guide to applied educational research using thematic analysis. *Practical Assessment, Research, and Evaluation* 25, 1 (2020), 2. doi:10.7275/ryr5-k983

Anne-Kathrin Peters, Mats Daniels, and Åsa Cajander. 2019. Utilising Diversity for Project Work and Learning: A Study of the Learning Agreement Intervention. In *2019 IEEE Frontiers in Education Conference (FIE)*. IEEE, New York, 1–9. doi:10.1109/FIE43999.2019.9028584

Kai Petersen, Sairam Vakkalanka, and Ludwik Kuzniarz. 2015. Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and software technology* 64 (2015), 1–18. doi:10.1016/j.infsof.2015.03.007

Jennifer A. Polack-Wahl. 2001. Enhancing group projects in software engineering. In *Proceedings of the Sixth Annual CCSC Northeastern Conference on The Journal of Computing in Small Colleges* (Middlebury, Vermont, USA) *(CCSC '01)*. Consortium for Computing Sciences in Colleges, Evansville, IN, USA, 111–121. https://dl.acm.org/doi/abs/10.5555/378613.378691

Joël Porquet-Lupine and Madison Brigham. 2023. Evaluating Group Work in (too) Large CS Classes with (too) Few Resources: An Experience Report. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1* (Toronto ON, Canada) *(SIGCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 4–10. doi:10.1145/3545945.3569788

Iwan Handoyo Putro, Angela Carbone, and Judy Sheard. 2014. Developing a framework to assess students' contributions during wiki construction. In *Proceedings of the Sixteenth Australasian Computing Education Conference - Volume 148 (ACE '14)*. Australian Computer Society, Inc., AUS, 123–131. https://dl.acm.org/doi/10.5555/2667490.2667505

Ser Wee Darren Quek, Jared Kai Fu Teo, Nasruddine Louahemmsabah, Yap Boon Cyrus Tong, Aaron Zheng Rong Poh, Qi Cao, Chee Kiat Seow, Peter Chunyu Yau, and Alex Qiang Chen. 2023. Improving Teamwork in Software Engineering Projects in Higher Education. In *2023 IEEE Frontiers in Education Conference (FIE)*. IEEE, New York, 1–8. doi:10.1109/FIE58773.2023.10343476

Phil Race. 1999. *2000 Tips for Lecturers* (1st ed.). Routledge, Milton Park, UK. doi:10.4324/9780203062357

Debbie Richards. 2009. Designing Project-Based Courses with a Focus on Group Formation and Assessment. *ACM Trans. Comput. Educ.* 9, 1, Article 2 (March 2009), 40 pages. doi:10.1145/1513593.1513595

Buturab Rizvi, Ebrahim Bagheri, and Dragan Gasevic. 2015. A systematic review of distributed Agile software engineering. *Journal of Software: Evolution and Process* 27, 10 (2015), 723–762. doi:10.1002/smr.1718

Seán Russell. 2025. Heuristic for Individual Grading in Group Software Projects. In *Proceedings of the ACM Global Computing Education Conference 2025 - Volume 2* (Gaborone, Botswana) *(CompEd 2025)*. Association for Computing Machinery, New York, NY, USA, 394–395. doi:10.1145/3736251.3747308

Sandra Schulz, Sarah Berndt, and Anja Hawlitschek. 2023. Exploring students' and lecturers' views on collaboration and cooperation in computer science courses-a qualitative analysis. *Computer Science Education* 33, 3 (2023), 318–341. doi:10.1080/08993408.2021.2022361

Michael James Scott, Bedour Alshaigy, Angela A. Siegel, and Mark Zarb. 2023. Managing Group Projects in Undergraduate Computing. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 2* (Turku, Finland) *(ITiCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 597–598. doi:10.1145/3587103.3594163

Michael James Scott, Steve Counsell, Stanislao Lauria, Stephen Swift, Allan Tucker, Martin Shepperd, and Gheorghita Ghinea. 2015. Enhancing practice and achievement in introductory programming with a robot olympics. *IEEE Transactions on Education* 58, 4 (2015), 249–254. doi:10.1109/TE.2014.2382567

Michael James Scott, Alcwyn Parker, Brian McDonald, Gareth Lewis, and Edward J. Powley. 2019. Nurturing Collaboration in an Undergraduate Computing Course with Robot-themed Team Training and Team Building. In *Proceedings of the 3rd Conference on Computing Education Practice* (Durham, United Kingdom) *(CEP '19)*. Association for Computing Machinery, New York, NY, USA, Article 5, 4 pages. doi:10.1145/3294016.3294019

Georg Simmel. 2009. *Sociology: Inquiries into the construction of social forms*. Brill, Leiden, NL.

Dominique MA Sluijsmans, George Moerkerke, Jeroen JG Van Merrienboer, and Filip J Dochy. 2001. Peer assessment in problem based learning. *Studies in educational evaluation* 27, 2 (2001), 153–173.

Bhavani Sridharan and David Boud. 2019. The effects of peer judgements on teamwork and self-assessment ability in collaborative group work. *Assessment & Evaluation in Higher Education* 44, 6 (2019), 894–909. doi:10.1080/02602938.2018.1545898

Bhavani Sridharan, Joanna Tai, and David Boud. 2019. Does the use of summative peer assessment in collaborative group work inhibit good judgement? *Higher Education* 77 (2019), 853–870. doi:10.1007/s10734-018-0305-7

Jan-Philipp Steghöfer, Håkan Burden, Regina Hebig, Gul Calikli, Robert Feldt, Imed Hammouda, Jennifer Horkoff, Eric Knauss, and Grischa Liebel. 2018. Involving External Stakeholders in Project Courses. *ACM Trans. Comput. Educ.* 18, 2, Article 8 (July 2018), 32 pages. doi:10.1145/3152098

Ivan Dale Steiner. 1972. *Group process and productivity*. Vol. 1. Academic press New York, New York. https://archive.org/details/groupprocessprod0000stei/page/n5/mode/2up

Ben Stephenson, Mark James, Nigel Brooke, and John Aycock. 2016. An Industrial Partnership Game Development Capstone Course. In *Proceedings of the 17th Annual Conference on Information Technology Education* (Boston, Massachusetts, USA) *(SIGITE '16)*. Association for Computing Machinery, New York, NY, USA, 136–141. doi:10.1145/2978192.2978214

Jacklin Stonewall, Michael Dorneich, Cassandra Dorius, and Jane Rongerude. 2018. A Review of Bias in Peer Assessment. In *2018 CoNECD - The Collaborative Network for Engineering and Computing Diversity Conference*. ASEE Conferences, Crystal City, Virginia, Article 24251, 9 pages. https://peer.asee.org/29510.

Anya Tafliovich, Andrew Petersen, and Jennifer Campbell. 2015. On the Evaluation of Student Team Software Development Projects. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (Kansas City, Missouri, USA) *(SIGCSE '15)*. Association for Computing Machinery, New York, NY, USA, 494–499. doi:10.1145/2676723.2677223

Saara Tenhunen, Tomi Männistö, Matti Luukkainen, and Petri Ihantola. 2023. A systematic literature review of capstone courses in software engineering. *Information and Software Technology* 159 (2023), 107191. doi:10.1016/j.infsof.2023.107191

The Joint ACM/IEEE/AAAI Task Force on Computing Curricula. 2023. *Computer Science Curricula 2023 Version Gamma*. Association for Computing Machinery (ACM) and IEEE Computer Society. Retrieved November 28, 2024 from https://csed.acm.org/wp-content/uploads/2023/09/Version-Gamma.pdf

The Joint Task Force on Computing Curricula Association for Computing Machinery (ACM) and IEEE Computer Society. 2013. *Computer Science Curricula 2013*. Association for Computing Machinery (ACM) and IEEE Computer Society. Retrieved December 16, 2023 from https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf

Keith Topping. 1998. Peer assessment between students in colleges and universities. *Review of educational Research* 68, 3 (1998), 249–276. https://www.jstor.org/stable/1170598

Marco Torchiano, Antonio Vetrò, and Riccardo Coppola. 2024. Teaching Scrum with a focus on efficiency and inclusiveness. In *Proceedings of the 28th International Conference on Evaluation and Assessment in Software Engineering* (Salerno, Italy) *(EASE '24)*. Association for Computing Machinery, New York, NY, USA, 595–599. doi:10.1145/3661167.3661277

Chia-Wen Tsai, Michael Yu-Ching Lin, Yih-Ping Cheng, Lan-Yu Lee, Wen-Li Chyr, Chih-Hsien Lin, Jian-Wei Lin, and Meng-Chuan Tsai. 2023. The effects of online peer-facilitated learning and distributed pair programming on students' learning. *Computers & Education* 203 (2023), 104849. doi:10.1016/j.compedu.2023.104849

Laura Tubino, Andrew Cain, Jean-Guy Schneider, Dhananjay Thiruvady, and Niroshinie Fernando. 2020. Authentic individual assessment for team-based software engineering projects. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Software Engineering Education and Training* (Seoul, South Korea) *(ICSE-SEET '20)*. Association for Computing Machinery, New York, NY, USA, 71–81. doi:10.1145/3377814.3381702

Laura Tubino, Jean-Guy Schneider, Andrew Cain, Dhananjay Thiruvady, and Chathu Ranaweera. 2021. Reforming Assessment: Challenges Beyond Design. In *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)*. IEEE, New York, 78–88. doi:10.1109/ICSE-SEET52601.2021.00017

Bruce Tuckman. 1965. Developmental sequence in small groups. *Psychological Bulletin* 63, 6 (1965), 384–399.

Roksana Jahan Tumpa, Samer Skaik, Miriam Ham, and Ghulam Chaudhry. 2022. A holistic overview of studies to improve group-based assessments in higher education: A systematic literature review. *Sustainability* 14, 15 (2022), 9638.

Marjo Van Zundert, Dominique Sluijsmans, and Jeroen Van Merriënboer. 2010. Effective peer assessment processes: Research findings and future directions. *Learning and instruction* 20, 4 (2010), 270–279.

Maria Vasilevskaya, David Broman, and Kristian Sandahl. 2014. An assessment model for large project courses. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education* (Atlanta, Georgia, USA) *(SIGCSE '14)*. Association for Computing Machinery, New York, NY, USA, 253–258. doi:10.1145/2538862.2538947

Maria Vasilevskaya, David Broman, and Kristian Sandahl. 2015. Assessing Large-Project Courses: Model, Activities, and Lessons Learned. *ACM Trans. Comput. Educ.* 15, 4 (Dec. 2015), 20:1–20:30. doi:10.1145/2732156

Brett Vaughan, Jacqui Yoxall, and Sandra Grace. 2019. Peer assessment of teamwork in group projects: Evaluation of a rubric. *Issues in Educational Research* 29, 3 (2019), 961–978.

Paul Hubert Vossen and Suraj Ajit. 2020. Towards a Paradigm Change in Group and Peer Assessment in Software Engineering Education. In *2020 IEEE 32nd Conference on Software Engineering Education and Training (CSEE&T)*. IEEE, New York, 1–10. doi:10.1109/CSEET49119.2020.9206197

Esther Su Wee Tan, Muhammad Ridhwan Bin Sani, Nor Atiqah Shahira B. Norshahid, Aloysius Weijun Loo, Tuan Ming Matthew Chuang, and Qi Cao. 2023. Teamwork Improvement in Group Projects of Professional Software Development. In *2023 3rd International Conference on Educational Technology (ICET)*. IEEE, New york, 244–248. doi:10.1109/ICET59358.2023.10424397

Jacqueline Whalley, Asanthika Imbulpitiya, and Tony Clear. 2025. Computing capstone courses as preparation for practice: A global survey of instructors. *ACM Inroads* 16, 1 (2025), 26–39.

Mattias Wiggberg. 2008. *Unwinding processes in computer science student projects*. Ph. D. Dissertation. Uppsala universitet. https://uu.diva-portal.org/smash/record.jsf?pid=diva2%3A116653&dswid=-7041

Mark Wilson and Kathryn Sloane. 2000. From Principles to Practice: An Embedded Assessment System. *Applied Measurement in Education* 13, 2 (2000), 181–208. doi:10.1207/S15324818AME1302_4

Dominic Wyse, Larry E Suter, Emma Smith, and Neil Selwyn. 2016. *The BERA/SAGE handbook of educational research*. SAGE Publications Ltd, London. 1–1170 pages. doi:10.1080/00071005.2019.1578063

Anita Yang, Anna Brown, Rachel Gilmore, and Adam M Persky. 2022. A practical review for implementing peer assessments within teams. *American Journal of Pharmaceutical Education* 86, 7 (2022), 8795.

Chang Yun, Hesam Panahi, and Zhigang Deng. 2016. A Multidisciplinary, Multifaceted Approach to Improve the Computer Science based Game Design Education: Methodology and Assessment. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education (SIGCSE '16)*. Association for Computing Machinery, New York, NY, USA, 570–575. doi:10.1145/2839509.2844582

Alexey Zagalsky, Joseph Feliciano, Margaret-Anne Storey, Yiyun Zhao, and Weiliang Wang. 2015. The Emergence of GitHub as a Collaborative Platform for Education. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing* (Vancouver, BC, Canada) *(CSCW '15)*. Association for Computing Machinery, New York, NY, USA, 1906–1917. doi:10.1145/2675133.2675284

Received October 1 2025; revised May 28 2026; accepted June 11 2025