International Journal of Engineering Education Vol. 41, No. 6, pp. 1675–1687, 2025
Printed in Great Britain

0949-149X/91 $3.00+0.00

Enhancing Learning and Deliverable Quality in Software Engineering through Oral Inquiry Audits*

PAULINE C. WADE
Department of Computer Science & Engineering, Texas A&M University, 3112 TAMU, Peterson Building, College Station, TX 77843, USA. E-mail: paulinewade@tamu.edu

HILLARY E. MERZDORF
Purdue University, West Lafayette, IN 47907, USA. E-mail: hmerzdorf@gmail.com

TRACY HAMMOND
Department of Computer Science & Engineering, Texas A&M University, 3112 TAMU, Peterson Building, College Station, TX 77843, USA. E-mail: hammond@tamu.edu

Industry-aligned quality assurance (QA) education remains underexplored in software engineering (SE) curricula, contributing to graduates' unpreparedness for QA roles due to misalignments between academic training and industry practices. This study examines the impact of Oral Inquiry Audits (OIAs) – a pedagogical tool derived from industry practices involving oral discussions on QA processes – on undergraduate SE students' group learning and project deliverable quality. Using a quasi-experimental design, SE students developing web applications for real clients within an ISO 25010:2023-aligned QA framework were divided into control (no OIAs) and treatment (OIAs) groups. Data collection included assessments of QA knowledge and customer feedback, with statistical analyses comparing results across groups. This study found that OIAs enhanced group learning and project deliverables in team-based settings, with the treatment group showing greater improvements than the control group, particularly in Integrity and Usability. OIAs can serve as a replicable academic tool to bridge academia-industry gaps by providing experiential, standards-aligned QA training, enhancing graduates' readiness for QA roles, and integrating ISO frameworks into SE education.

Keywords: industry–academia alignment; oral inquiry audits; pedagogy; quality assurance; software engineering; software quality; undergraduate education; zone of proximal development

1. Introduction

A comprehensive quality assurance (QA) approach is critical for producing high-quality software products. This study adopts the ISO/IEC 25010:2023 standard, which defines key characteristics of software quality – in the categories of compliance, correctness, maintainability, usability, and integrity – relevant to applications developed by students in a software engineering (SE) course [1–3]. These characteristics serve as benchmarks for ensuring software meets user needs.

QA processes are integrated throughout the software development lifecycle (SDLC), beginning with requirements analysis and extending through testing and maintenance phases. Activities include requirement analysis, risk management, validation test planning, code reviews, and various types of testing (e.g., unit, integration) [3, 4]. The goal is to prevent defects, detect issues early, and improve SE practices to deliver high-quality products [1, 2].

Inconsistent application of QA processes can lead to software quality failures due to varying compliance levels or inconsistent standards among team members [4, 5]. This study focuses on addressing these inconsistencies within individual student projects

1.1 Statement of the Problem

QA plays a critical role in software development, yet there is a persistent misalignment between academic instruction and industry practices in SE education [6-8]. This gap contributes to inadequate QA implementation, leading to software failures, increased maintenance costs, and diminished quality [9, 10]. Addressing this issue requires training on quality principles and preventive measures throughout the software lifecycle [9, 10, 11–13].

This study investigates Oral Inquiry Audits (OIAs), an industry-derived pedagogical tool designed to enhance QA education by aligning academic practices with industry standards. OIAs supplement documentary audits by incorporating interactive discussions where team members explain their QA processes and rationale, providing real-time feedback and fostering deeper understanding of QA principles [3, 14, 15]. Grounded in Vygotsky's zone of proximal development (ZPD), OIAs leverage social learning through interactions with auditors and peers, accelerating comprehension and improving communication skills essential for professional QA roles [16, 17].

In SE education, OIAs can be used to evaluate compliance with organizational QA processes by examining artifacts such as test plans and test

* Accepted 25 August 2025.

results. They align with ISO/IEC 25010:2023 standards and Capability Maturity Model Integration (CMMI) Level 5 principles to ensure consistency and foster iterative refinement of quality goals [1, 2]. For example, auditors may assess usability by examining evidence of learnability metrics – defined as the ability of users to learn product functions within a specified timeframe – and verifying adherence to prescribed processes [1].

By contributing to existing pedagogical tools which can help bridge the gap between academia and industry, OIAs can improve both learning outcomes and project deliverable quality. Students gain hands-on experience applying QA processes in real-world scenarios while developing skills in teamwork, reflection, and communication [14, 15]. This innovative approach addresses key challenges in current QA pedagogy – such as limited alignment with industry standards [3, 6, 18] – and positions OIAs as a replicable framework for enhancing SE education.

1.2 Purpose and Research Questions

This study explores the use of OIAs as a pedagogical approach in QA education in the SE domain, as a tool to increase alignment between academic instruction and industry practices. This is based on the premise that exposing students to industry-like processes in the classroom has the potential of increasing alignment with industry practices, possibly enhancing students' employability and readiness to enter the workforce [18].

Specifically, the purpose of this study is to investigate the impact of OIAs on (a) group learning of QA concepts, and (b) correlation of learning with project deliverable quality. In the context of the study, OIAs help ensure adherence of artifacts against the SE process framework provided to students (documented in the given project notebook), which is inspired by ISO and CMMI standards. Adherence is assessed using a rubric, standardized scripts, and protocols, which are aligned with selected quality characteristics (applicable for web applications developed in the class) as defined by the ISO/IEC 25010:2023 [1] and CMMI [2] standards; and inspired by QA principles described by Pressman & Maxim [3]. This study determined the impact of OIAs in aligning project deliverables with project requirements as agreed upon with the assigned real customers.

This study was guided by the following research questions:

Research Question 1: How does the use of OIAs impact learning of QA, as a group, as measured by pretest and posttest assessments?

Research Question 2: What is the correlation between learning QA through the use of OIAs and project deliverable quality?

2. Literature Review

This literature review explores the current state of QA education in SE, focusing on the use of OIAs as a pedagogical approach to better align academic instruction with industry demands. It provides an overview of QA concepts, challenges in aligning education with industry standards, and innovative teaching methods, including audits and peer reviews, while situating the study within its theoretical framework.

2.1 Quality Assurance

QA, an integral component of a QMS, is a systematic, planned set of actions designed to promote high-quality products, services, and processes while promoting continuous improvement [3, 19, 20]. Within this framework, QA focuses on process-oriented, preventive actions; in contrast, QC emphasizes product-focused, detective actions [21–23]. Together, QA and QC provide leading and lagging indicators for comprehensive performance management [1–3, 24, 25].

The International Organization for Standardization 9001:2000 (ISO 9001:2000) and CMMI define essential QMS elements applicable across disciplines [2, 24]. Specifically, ISO 9001:2000 for engineering organizations offers a framework for defining, identifying, and managing interrelated processes to enhance overall quality, ultimately leading to the fulfillment of customer requirements [1, 3, 24]. CMMI is a process improvement framework, developed by the Software Engineering Institute at Carnegie Mellon University, designed to help organizations enhance their development processes and deliver high-quality products and services. [2].

Researchers have found inadequate QA can result in increased failures, higher maintenance costs, and decreased software quality [9, 11, 12]. In fact, despite the importance of QA and QMS [3], there remains a pressing need to align academic instruction with industry practices in QA. To address this issue, researchers have suggested the need for innovative pedagogical approaches to bridge the gap between theoretical knowledge and practical application of QA principles. One such approach proven effective in industry is the use of audits [3], which has been largely unexplored as a learning tool in academic settings.

2.2 Industry–Academia Alignment in Quality Assurance Education

Acharya [6] highlighted the need to align SV&V

education with industry practices for engineering graduates. Wilson and Johnson [26], in their study of the Australian IT industry noted a growing interest in quality management, and a university's response by adapting its educational offerings to respond to industry trends. Stepanova et al. [27] in their paper on employer feedback on hiring computer science students, highlighted discrepancies between undergraduate computer science program outcomes and employers' expectations, particularly in fostering soft skills and practical skills, including QA. Employers advocate for a curriculum that teaches students a disciplined software development approach which includes practices that produce high-quality results.

These studies have identified challenges in current QA education, including insufficient hands-on exercises and real-world applications in SV&V education; lack of exposure to industry standards such as ISO 9001, IEEE, and CMMI; and a shortage of trained SV&V practitioners in industry attributable to ineffective pedagogy. Before implementing QA, organizations should provide comprehensive training on quality principles and goals, corresponding policies and processes, relevant metrics, and monitoring techniques for consistent application [13]. This training is considered crucial for ensuring effective implementation and consistent application of QA practices across an organization.

2.3 Current Pedagogy in Quality Assurance

Researchers have explored various pedagogical approaches to teaching QA, highlighting both strengths and limitations. Pádua [28] demonstrated the effectiveness of audits in improving project deliverable quality but noted the lack of control groups or comparative analyses, limiting the assessment of this approach. Similarly, Laporte and April [29] integrated lectures, practical exercises, and team projects but did not present quantitative data or comparative evaluations. Other studies, such as Jenkins [30] and Thompson and Edwards [31], relied on observational insights rather than empirical data, while Acharya et al. [32] introduced hands-on activities like LEGO-based exercises and active learning tools but lacked rigorous evaluation of learning outcomes. Elgrably and Oliveira [33] conducted a quasi-experimental study comparing active learning with traditional methods, finding improved outcomes for the experimental group but excluding audits from their model. Across these studies, common challenges persist, including insufficient empirical evaluation, limited alignment with industry standards, and a focus on defect detection rather than prevention [29, 34–36]. These gaps highlight the need for innovative approaches like OIAs, which emphasize process consistency and industry alignment to address these pedagogical challenges effectively.

2.4 Audits

An audit is a systematic, independent examination of an organization's records, processes, or systems to verify their accuracy, effectiveness, and compliance with established standards or regulations [37]. This formal evaluation serves as a critical tool for ensuring accountability, improving quality, and identifying areas for enhancement across various domains, including healthcare, business, research, and SE.

Audits have demonstrated significant benefits across various fields, such as improving clinical outcomes and patient safety in healthcare [38, 39], enhancing compliance with counseling frameworks in social services [40], and increasing financial transparency and credibility in business [37, 41]. In qualitative research, audit procedures have been shown to help maintain consistent quality in research results and increase accountability [42]. The authors found the audit process aligned with social learning principles in terms of social interaction, observation, and modeled behaviors. This alignment improved attention and retention, boosted motivation, and enhanced learning through others' experiences.

These findings across diverse fields highlight the potential of audits as a possible tool for quality improvement and learning, suggesting their potential value in SE education. However, although audits have shown promise in various fields, it is important to compare them with other pedagogical approaches used in engineering education.

2.5 Comparison with Other Pedagogical Approaches

Several pedagogical approaches in engineering education share similarities with audits but lack the comprehensive features of OIAs. Cooperative learning emphasizes teamwork and accountability but does not include iterative reflection or alignment with industry standards, which are central to OIAs [43]. Collaborative learning fosters peer interaction and reflection but lacks structured evaluation and external feedback, which OIAs provide through standardized audits and debriefing sessions [44, 45]. Peer reviews focus on product quality but overlook process compliance, whereas OIAs evaluate adherence to team-wide QA processes, promoting continuous improvement [3, 46, 47]. Grounded in ISO/IEC 25010:2023 and CMMI Level 5 principles, OIAs integrate social learning and iterative refinement, bridging academic instruction with industry practices to enhance SE education [2, 3].

2.6 Theoretical Framework

Vygotsky's ZPD provides the theoretical foundation for exploring the impact of OIAs on learning QA through social interactions within student teams and between students and auditors. According to Vygotsky's ZPD, learning occurs most effectively when a more knowledgeable individual guides students through the learning process within their developmental zone [16, 17]. Although originally developed in the context of childhood development, the ZPD concept has been applied successfully to other domains, including engineering education [48].

In this study, OIAs facilitate interactions among team members during audit preparation, with auditors during the audit, and in post-audit discussions. These interactions align with Vygotsky's emphasis on social learning, where guidance from more knowledgeable individuals – whether peers or auditors – accelerates students' understanding of QA processes. By requiring students to articulate their understanding of QA practices verbally, OIAs promote real-time feedback and reflection, helping students identify and address gaps in their comprehension that might not be evident through written documentation alone. Additionally, these interactions foster essential communication skills [14, 15] critical for professional QA roles.

Grounded in Vygotsky's ZPD theory, this research examines how OIAs influence both group learning outcomes and project deliverable quality in an undergraduate SE class. The study hypothesizes that the social learning fostered by OIAs enhances students' ability to apply QA concepts effectively while improving the quality of their project deliverables as measured by user acceptance testing results and customer satisfaction [3, 49].

3. Methods

This study investigated the impact of OIAs on group learning of QA, and correlation with project deliverable quality as part of an undergraduate SE class. Vygotsky's ZPD [16, 17] form the basis for exploring the effect of OIAs on group learning and the hypothesized impact to project deliverable quality. According to the ZPD, specifically in the area of social learning, learning happens when a more knowledgeable person [16, 17], represented in this study by other team members and the auditor, guides the students through the learning process. Guided by Creswell and Guetterman [50], this section provides an overview of the research methodology and the pedagogical approach to address the study's research questions.

3.1 Overall Research Design

This study employed a quasi-experimental quantitative approach to compare control and treatment groups, using the nonequivalent control group design [51]. Both control and treatment groups consisted of project teams tasked with developing web applications for real customers who represent organizations with needs for automation or decision support. Intact groups in this study, represented by class sections, are used rather than random assignment of individuals, and pretests and posttests are administered to both groups. This design was chosen given the constraints of the setting of the class, where random assignment of individual students is difficult due to the possible wide disparity of collective skill level across control and treatment groups. It allows for a comparison between groups while acknowledging the potential preexisting differences between the intact groups. Sections were randomly assigned to control or treatment groups. The study includes measures for both groups to assess the impact of the OIAs on group learning, and its correlation with project deliverable quality.

Both groups used the same experiential-learning curriculum, a learner-centered pedagogy that emphasizes learning-by-doing [52].Statistical analysis was used to assess differences between the control and treatment groups across two dependent variables: learning and project deliverable quality. The study was conducted over multiple semesters, with group learning, and its effect on project deliverable quality, being investigated simultaneously. This research design aims to provide an evaluation of the impact of OIAs when combined with experiential-based learning

3.2 Pedagogical Approach

The pedagogical approach in this study aims to achieve two key learning outcomes: equipping students with the ability to articulate relevant ISO-based quality characteristics and corresponding targets, and with the ability to describe the processes and artifacts that contribute to meeting these quality characteristics. The approach is grounded in experiential learning (EL), a learner-centered pedagogy that bridges classroom theory with industry skills by engaging students in active problem-solving and reflection [52]. EL fosters deeper understanding of engineering concepts, project management, and professional collaboration while aligning educational outcomes with workforce demands. In this study, students were divided into project teams using stratified randomization to ensure balanced representation based on shared characteristics, such as QA experience [53]. Each team was assigned a real-world customer requiring a web-based application for automation or decision support. Teams followed an SE process framework inspired by ISO/IEC 25010:2023 [1] and CMMI standards [2], which demonstrated a 97% success rate in previous student projects [1, 2, 24]. A rubric aligned with the process framework guided students through project execution. Development occurred over three iterative sprints, short phases typical of agile projects that adapt to changing requirements [3].

The treatment group underwent OIAs after each sprint, while the control group followed the standard curriculum without audits. OIAs were inspired by Gantz's IT audit methodology and conducted by teaching assistants using standardized prompts and rubrics [54]. During OIAs, auditors assessed adherence to QA processes by examining artifacts and engaging students in discussions about quality characteristics such as usability. For example, auditors evaluated learnability – a sub-characteristic of usability defined as the ability of users to learn product functions within a specified timeframe – by reviewing metrics and evidence of process adherence [1]. This hands-on approach aligns academic instruction with industry practices while fostering reflection and skill development critical for professional QA roles. By integrating EL principles with OIAs, this study provides a replicable framework for teaching QA concepts in SE education.

3.3 OIA Implementation

Each OIA consisted of a structured, interactive dialogue between the student development team and an assigned auditor and was conducted once after each sprint (for a total of three audits) [56]. OIAs supplemented documentary audits of project artifacts by adding a dynamic discussion layer, promoting deeper engagement with QA principles.

The auditor's primary role was to verify each team's adherence to the course's SE process framework – as documented in their project notebook – and to prompt students to clearly articulate and justify their QA decisions. Both the process framework and accompanying rubrics were aligned to recognized standards, including ISO/IEC 25010:2023 quality characteristics and relevant CMMI practices. Below is a description of a session sequence:

1. *Presentation of Artifacts:* The team began by presenting key artifacts from the sprint, such as updated user stories, acceptance criteria, test case documentation, and usability test results.
2. *Rubric-Guided Inquiry:* The auditor initiated a series of questions from an audit rubric tailored to the QA characteristic under review, such as Usability, Compliance, Integrity, Maintainability, or Correctness. For instance, when reviewing Usability, teams were asked to provide evidence supporting specific sub-characteristics like Learnability or Inclusivity. For more detail on the audit rubrics and evaluation criteria, see the "Assessment Instruments" section.
   - Example (Learnability): The auditor looked for requirements that were both measurable and testable (e.g., "A college-aged user shall perform Task A in under five minutes").
   - Teams could be prompted: "Please show evidence that users can accomplish Task A within the specified time."
   - Example (Inclusivity): The auditor requested evidence that the application's accessibility had been evaluated according to the Web Content Accessibility Guidelines (WCAG), potentially using a WAVE report to identify any compliance errors or usability barriers for people with disabilities.
3. *Evidence and Rationale:* The auditor requested supporting documentation (such as test coverage reports, user feedback, design mock-ups) and probed for the rationale behind the team's QA decisions. This approach encouraged students to reflect on both procedural and user-centered aspects of their software.
4. *Real-Time Response and Dialogue:* Student teams responded in real time, explaining their reasoning, referencing relevant metrics, and acknowledging any deviations from the prescribed process. The interactive nature of the audit facilitated immediate clarification and fostered student growth through direct feedback.
5. *Feedback and Incremental Improvement:* The auditor provided timely feedback, highlighting both areas of compliance and any deficiencies requiring correction prior to the next sprint. This iterative cycle reinforced continuous improvement and ensured that students had specific, actionable goals for subsequent work.

To provide further context for these procedural steps, students and customers shared their experiences with the OIA process. Several student participants noted that the structured audit questioning in OIAs "prompted us to revisit test cases before each sprint review," enabling their teams to catch errors early and refine deliverables. Customers consistently noted that teams "asked thoughtful questions that clarified expectations and technical specifications," and by Sprint 3 observed "visible gains in visual consistency, user flow optimization, and overall application quality." One student reported that towards the end of the software development lifecycle "our customer felt the UI was intuitive, even similar to existing software he uses," underscoring how iterative, rubric-guided feedback not only improved technical correctness but also aligned products with professional usability standards.

3.4 Site Location and Population

The study was conducted at a large Southwestern R1 HSI university. Participants are students in the Computer Science & Engineering BS program enrolled in an SE course. Students in the SE course have diverse backgrounds, including varying levels of prior relevant work experience. Many have a high GPA, reflecting the competitive entry threshold for Computer Science and Engineering. This location was selected through convenience sampling, leveraging the researcher's employment at the institution and role as the class instructor for the class. This approach was chosen primarily due to ease of access to the student population, time and cost efficiency, and the researcher's familiarity with the organizational environment [51]. The location provided an opportunity to assess the impact of OIAs on students' learning in QA, correlated with project deliverable quality, in a real classroom setting.

The population consisted of all students enrolled in an SE course, consisting of junior and senior level undergraduate students, mostly majoring in computer science, with a small percentage majoring in computer engineering and other disciplines such as electrical engineering. In an academic year, there are between 100 and 150 students enrolled in this class. The SE class is an elective course with the prerequisite of the foundations of SE course for students majoring in Computer Science and Engineering. However, this pre-requisite is not mandatory for students outside of Computer Science and Engineering. This SE course is the only one conducted in the spring and fall semesters.

3.5 Variables

The independent variable for this experiment was the OIAs administered to the treatment group participants. The dependent variables measured were learning of QA and project deliverable quality. To minimize intermingling, sections were randomly chosen to contain only control or only treatment groups. Both groups received identical course materials, pretest and posttest instruments, and instruction from the same instructor and teaching assistants whenever possible. Teaching assistants assigned to the treatment group were provided with standardized OIA protocols, scripts, and rubrics to maintain consistency. The treatment group were subjected to OIAs, while the control group followed the standard curriculum without modifications. Finally, posttest measures were administered to both groups to evaluate the effectiveness of OIAs on learning outcomes and deliverable quality, enhancing the study's internal validity and reliability.

3.6 Data Collection

Data collection used quantitative methods, consisting of pretests and posttests using audit rubrics and customer feedback forms. Pretests, consisting of audit rubrics, were given during the first of three iterative sprints, short phases typical of agile projects and posttests after the third and last sprint. Audit rubrics assessed group adherence to QA processes during sprints, incorporating principles from CMMI Level 5 [2] and ISO/IEC 25010:2023 [1]. For example, students were asked to describe the monitoring of quality objectives and the process they applied to achieve them. Customer feedback forms inspired by Hill et al. [55] assessed satisfaction with deliverables.

Stratified randomization balanced participants' prior experience across treatment and control groups. Treatment groups underwent OIAs using standardized protocols, while control groups simply followed the standard curriculum. Data was gathered at multiple points – after each sprint. Creswell and Creswell's [51] exclusion criteria addressed incomplete data, nonparticipation, outliers, inter-group collaboration, attrition, and prior audit experience. Identical course materials and trained personnel ensured comparability across groups.

This study prioritized bias mitigation to enhance internal validity, as emphasized by Creswell and Creswell [51]. Key strategies included:

Standardized Training: Instructors and teaching assistants underwent uniform training to ensure consistent application of OIA procedures.

- Standardized Grading: Detailed rubrics with clear criteria guided evaluations. Graders participated in training sessions, mock evaluations, and regular discussions to address discrepancies, ensuring consistent grading standards.
- Blind Methodology: Bias was minimized through measures such as auditors with minimal student interaction, anonymized data coding during analysis, and blind evaluation of deliverables and results.
- Standardized Protocols: Uniform scripts and procedures were employed to reduce subjective interpretations during OIAs.

These measures collectively aimed to control bias throughout data collection and analysis, to help bolster the study's reliability.

3.7 Assessment Instruments

Below is a description of the assessment instruments used in this study: the group audit rubrics employed to measure group learning (with an illustrative example for Usability) and the instrument used to gather customer feedback on project deliverable quality.

3.7.1 Assessment Instrument for Usability (Group Audit Rubric)

To systematically assess group learning and deliverable quality in usability, we employed a structured group audit rubric aligned with industry standards. This rubric was applied during OIAs to evaluate the presence, quality, and effectiveness of usability-focused practices and artifacts in student projects. The criteria reflect ISO/IEC 25010:2023 usability characteristics and established quality assurance methodologies, encouraging evidence-based reflection, measurable outcomes, and continuous improvement.

Each team's work was evaluated using weighted criteria contributing to a composite score out of 100%. Evaluators used a four-point scoring guide to ensure that assessments were specific and detailed, minimizing the risk of generic or off-topic responses.

Usability Assessment Criteria and Weights:

1. Usability Requirements (20%)
   - Evidence of user stories and/or acceptance criteria explicitly addressing usability.
   - Requirements specified as measurable and testable (e.g., "A college-aged user shall be proficient in basic system use within five minutes").
2. Test Case Evidence – Sunny-Day and Rainy-Day Scenarios (20%)
   - Test cases cover both typical (sunny-day) and edge/failure (rainy-day) user scenarios.
   - Test documentation includes clear inputs, expected outcomes, and pass/fail criteria.
3. Usability Test Coverage (15%)
   - Percentage of usability-related test cases executed, with documentation of pass/fail rates.
4. Usability Test Results (25%)
   - Representative samples of test results for usability scenarios, including both successes and failures.
5. Usability Monitoring Evidence (20%)
   - Ongoing documentation of usability tracked via project metrics addressing:
     - Learnability
     - Operability
     - User error protection
     - Inclusivity

   Scoring Guide:

   3 – Meets Expectations: Specific, detailed, and personally developed evidence; thoroughly addresses all criteria.

   2 – Partially Meets Expectations: Includes some specificity but relies on generic or template-based elements.

   1 – Minimally Meets Expectations: Mostly generalized and lacks sufficient detail.

   0 – Does Not Exist or Unacceptable: Off-topic or missing evidence.

This standardized rubric allowed the research team to compare both the breadth and quality of usability assurance across teams. By focusing on explicit artifacts and ongoing process monitoring, the instrument supports the alignment of student practices with professional expectations for usability-focused QA.

3.7.2 Customer Feedback Form and User Acceptance Testing (UAT)

At the conclusion of each sprint, actual customers completed a combined User Acceptance Testing (UAT) and customer satisfaction survey instrument adapted for agile practice. During UAT, critical deliverable features were evaluated using a pass/fail metric for key requirements, and overall satisfaction with features was rated on a five-point Likert scale. Customer data were collected blind to team assignment (customers were not informed whether teams were in the OIA or control group).

The customer feedback instrument included targeted items assessing both completeness and quality of project delivery:

- Completeness: Customers rated their satisfaction with how well the team addressed and delivered all agreed-upon requirements, using a point scale (0–3+). Survey prompt included: "Completeness – Rate your satisfaction on how well the team understood your requirements, as shown by the delivery of all agreed-to requirements." Branching follow-up questions prompted respondents to provide supporting information for their rating.
- Quality: Customers assessed how well the team implemented the specified features (as referenced in the UAT Form), including whether the application was free from errors and whether each feature met their definition of success. Satisfaction was recorded using the same rating scale, with open response fields for justification. Survey prompt included: Rate your satisfaction on how well the team implemented the agreed-to features described in the User Acceptance Test Form. Was the app free from errors? Did each feature satisfy your definition of success?"

By integrating both quantitative ratings and qualitative feedback, the customer satisfaction instrument provided detailed insights on functional delivery and implementation quality, supporting robust post-sprint evaluation of student projects.

3.8 Data Analysis by Research Question

3.8.1 Research Question 1: How does the use of OIAs impact group learning of QA?

The null hypothesis (H_{10}) posits no impact of OIAs on QA learning, while the alternative hypothesis (H_{1a}) suggests a positive effect. Data from OIA rubrics were analyzed. The Mann-Whitney U test [57], ideal for small samples and non-normal distributions, compared Sprint 1 and Sprint 3 results between treatment and control groups, providing a robust evaluation of OIAs' impact on group learning of QA.

3.8.2 3.8.2 Research Question 2: What effect do OIAs have on project deliverable quality?

The null hypothesis (H_{20}) posits no effect of OIAs on deliverable quality, while the alternative (H_{2a}) suggests a positive impact. Customer feedback forms inspired by Hill et al. [55] measured satisfaction ratings with teams developing identical applications for comparability. Statistical analysis via one-way and repeated measures ANOVA compared quality improvements (Sprint 1 vs. Sprint 3) between groups, supplemented by Levene's and Shapiro-Wilk tests to ensure validity despite non-normal data and small samples, providing a standards-aligned evaluation of deliverable quality.

With the study design, data collection protocols, and analytic approach detailed above, we next report the results of these analyses. Findings are presented in alignment with the research questions, focusing first on group learning gains and then on difference in project deliverable quality.

4. Results

This section presents the findings of our study on the impact of OIAs on group learning and project deliverable quality in an SE course. We analyzed data collected from control and treatment groups using various statistical methods, including one-way ANOVA, and repeated-measures ANOVA. Our analysis focused on comparing improvements in quality metrics between Sprint 1 and Sprint 3 for both groups and examining the overall effect of OIAs on group learning and customer satisfaction. The following subsections detail our findings, starting with descriptive statistics and progressing through more complex analyses.

4.1 Descriptive Statistics

We measured score improvement on five OIA metrics between Sprint 3 and Sprint 1 for control and treatment groups. The treatment group had wider improvements than the control group on all metrics except Compliance, where both groups had relatively little improvement (Treatment: M = 0.433, SD = 1.268; Control: M = 0.5, SD =1.225). The treatment group had the most score improvement for Usability (M = 1.275, SD = 1.419) and Integrity (M = 1.25, SD = 0.814). The control group also had the largest improvement on Usability (M = 0.875, SD = 0.386). Fig. 1 shows the mean score improvement between sprints for both groups.

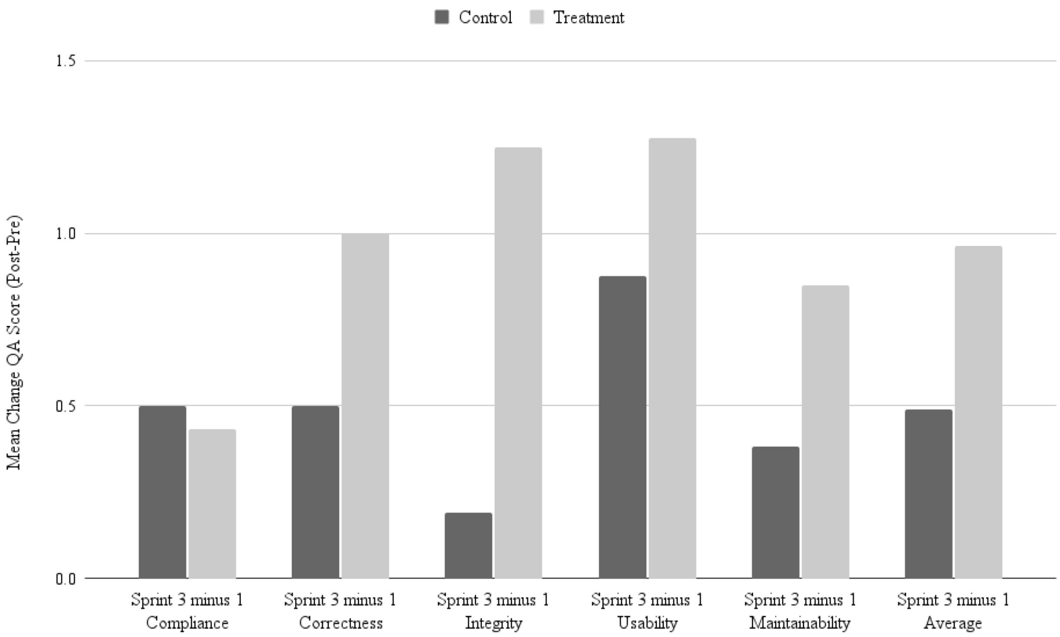

Fig. 1. Mean score improvement for control and treatment groups.

Table 1. Sprint 3 minus Sprint 1 t-tests between control and treatment

	t	df	p
Compliance	0.093	9.988	0.536
Correctness	–0.789	9.118	0.225
Integrity	–3.161	5.142	0.012*
Usability	–0.666	5.735	0.266
Maintainability	–1.692	8.098	0.064
Average	–1.217	7.433	0.13

Note. * = p < 0.05.

4.2 Mann-Whitney U Tests

We conducted a series of Mann-Whitney U tests, to determine whether the mean score improvements of Compliance, Correctness, Integrity, Usability, and Maintainability from Sprint 3 and Sprint 1 for the treatment group were significantly different from the control group (see Table 1 for detailed t-test results). We also tested the change differences for the average of all metrics. A significant difference between treatment and control groups was observed for Integrity $t(9.99) = -3.161$, $p = 0.012$. No other differences were significant, but Maintainability approached significance.

4.3 ANOVA

We performed a series of one-way analyses of variance (ANOVA) to examine the differences in improvements (delta) between control and treatment groups following the classroom intervention on the final project ratings of Usability, Integrity, Maintainability, Compliance, and Correctness. Levene's test of equality and Shapiro-Wilk's normality checks were performed for each test. We found that Integrity, $F(1, 10) = 5.174$, $p = 0.046$, and Usability, $F(1, 10) = 9.1$, $p = 0.013$, violated Levene's test due to unequal variance between groups, and Compliance and Integrity did not meet the Shapiro-Wilk test of normality ($W = 0.539$, $p > 0.001$; $W = 0.843$, $p = 0.03$).

Table 2. Sprint 3 minus Sprint 1 ANOVA results for control and treatment groups

		SS	df	MS	F	p	η^2
Compliance	Group	0.013	1	0.013	0.009	0.928	8.576×10-4
	Residuals	15.533	10	1.553			
Correctness	Group	0.75	1	0.75	0.622	0.449	0.059
	Residuals	12.06	10	1.206			
Integrity	Group	3.36	1	3.36	9.994	0.01*	0.5
	Residuals	3.362	10	0.336			
Usability	Group	0.48	1	0.48	0.444	0.52	0.043
	Residuals	10.813	10	1.081			
Maintainability	Group	0.653	1	0.653	2.861	0.122	0.222
	Residuals	2.283	10	0.228			

Note: * = p < 0.05.

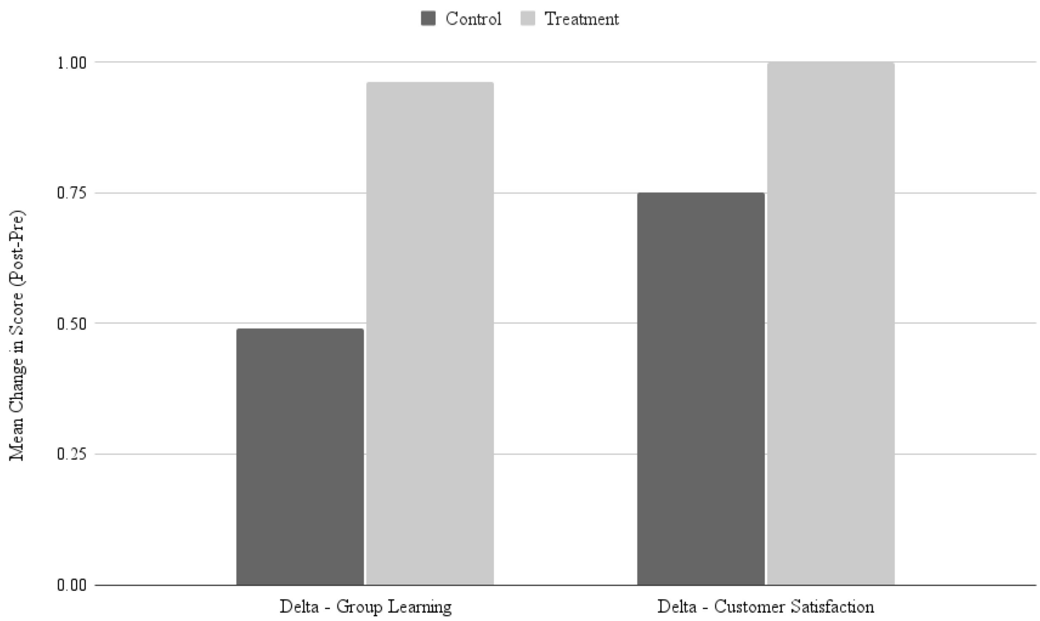

Fig. 2. Mean score differences for group learning metrics.

A significant difference between groups in mean score change (delta) was found for Integrity, $F(1,10) = 9.994$, $p = 0.01$, $\eta^2 = 0.5$ (see Table 2 for complete ANOVA results). Mean differences also approached significance for Maintainability, $F(1,10) = 2.861$, $p = 0.122$, $\eta^2 = 0.22$ at $\alpha = 0.10$. No other mean score differences were significant.

4.4 Repeated-Measures ANOVA

We performed a series of one-way repeated-measures analyses of variance (ANOVA) to determine whether quality of deliverables scores (Group Learning and Customer Satisfaction) were significantly different for control and treatment groups between Sprint 1 and Sprint 3.

Results of the repeated measures ANOVAs evaluating score improvements between groups showed a main effect of group on scores, $F(2, 20) = 5.338$, $p = 0.014$, $\eta^2 = 0.26$, indicating that scores were significantly different between control and treatment (see Fig. 2 for mean score differences for group learning metrics). However, there was no significant interaction between the repeated measures factor and group, $F(1, 10) = 2.164$, $p = 0.172$, $\eta^2 = 0.041$.

5. Discussion

The results of this study provide quantitative and practical evidence that OIAs can improve group learning of QA processes and, in certain domains, enhance project deliverable quality. Consistent with Vygotsky's ZPD, which posits that learning occurs most effectively when guided collaboration challenges learners just beyond their current independent capability, the OIA format positioned the auditor as a "more knowledgeable other" [16, 17] guiding student teams through a scaffolded reasoning process. Structured questioning and requests for evidence required teams to operate just beyond their current mastery, encouraging active resolution of gaps and aligning with prior findings in social learning contexts where immediate, targeted feedback accelerates conceptual understanding [43, 45].

In this study, auditors and peers prompted students to articulate their reasoning, defend QA decisions, and connect evidence to standards-based criteria. Over successive sprints, students shifted from heavy reliance on auditor guidance to independently applying QA principles – an indicator of progression through the ZPD. This iterative, socially mediated process advanced not only technical understanding but also professional competencies such as evidence-based justification, critical reasoning, and reflective self-assessment [48, 49].

The observed improvements in Integrity and Usability echo Pádua's [28] report of improved deliverable quality through audit-based pedagogy, while extending their work through the inclusion of a control group and systematic measurement of learning outcomes. Similarly, Laporte and April [29] integrated QA practices into project-based work but without the oral defense component which, in our study, appeared to solidify learning through articulation and justification. This aligns with findings from Kolmos and Holgaard [58], who reported that in team-based oral project examinations, the dialogic questioning process progresses from broad to deep technical inquiries, enabling students to demonstrate both breadth and depth of understanding. They found that such oral defense settings mirror professional engineering practice, fostering collaborative reasoning, communication skills, and the ability to articulate and defend decisions – skills highly valued by industry. Unlike Pádua [28] and Laporte and April [29], who reported only qualitative gains, this study provided statistically significant evidence by comparing gains of the treatment group (with OIAs) to those of the control group (standard EL pedagogy).

The use of OIAs encourages standardization of educational QA frameworks, grounded in industry standards such as ISO/IEC 25010:2023. This standard provided defined expectations for artifacts, processes, and evaluation criteria, which – when coupled with OIAs – minimized subjective interpretation by instructors and auditors and led to more consistent application of QA processes. By making these expectations explicit, OIAs helped ensure equal opportunities for project teams to succeed. This effect held regardless of differences in prior experience, skill mix, or work style. Social learning theory suggests that such a structured, equitable environment supports more uniform mastery of QA competencies. [45].

Beyond learning outcomes, customer satisfaction improved in both control and treatment groups. The use of OIAs exposes students to standardized and stable QA practices in academic settings – a practice important to workplace readiness [3]. This practical uniformity, even in the absence of large mean differences in customer satisfaction ratings, represents a noteworthy pedagogical contribution: graduates are better equipped to enter industry better prepared to follow and adapt formal QA procedures, apply structured methods, and engage in reflective, evidence-driven decision-making.

While these findings demonstrate important educational benefits, they must be interpreted in light of several methodological and practical limitations.

6. Limitations and Future Research

This study acknowledges several limitations that may have influenced its findings. Data collection relied on structured surveys and rubrics rather than richer qualitative approaches – such as focus group interviews – which could have illuminated the interactional dynamics that make OIAs effective. Researcher bias, stemming from prior experience with OIAs, may have introduced subjectivity, and customer evaluations of project deliverable quality may lack objectivity. The study's focus on undergraduate SE students limits its generalizability to other contexts, and the small sample sizes reduced statistical power, while the one-semester period posed challenges for students to fully grasp QA principles amidst competing course topics. Additionally, implementing OIAs required significant time, resources, and consistent application by teaching assistants, presenting logistical hurdles. Confusion between Compliance (alignment with customer requirements) and Correctness (alignment with developer interpretations) contributed to rating variability, likely due to students' limited exposure to requirement negotiation processes. Ceiling effects in customer satisfaction scoring may also have masked subtler differences in deliverable quality, suggesting the need for more granular quality metrics.

These kinds of challenges are not unique to this study. Consistent with findings reported by Drobnič Vidic [59], the use of process- and outcome-oriented assessment instruments – such as audit rubrics and authentic project portfolios – enhances the alignment between evaluation methods and intended learning outcomes. Nevertheless, limitations in reliability, student workload, and formative feedback echo challenges in the broader engineering education literature. Future research should continue to refine these instruments and study their impact on skill development and transferability in professional contexts.

Despite these limitations, the study showed that OIAs significantly enhanced group learning in domains such as Integrity and Usability, as well as project deliverable quality – gains attributed to OIAs' emphasis on social learning and iterative customer feedback. However, non-significant findings in Maintainability highlight the challenges of measuring long-term code sustainability in short academic projects; future studies should consider phased assessments mirroring industry workflows, such as modularity evaluations, code reviews, and post-delivery testing.

To address these limitations, future research should incorporate qualitative methods and individual-level analyses using covariates such as extracurricular experiences and should expand the intervention across more class sections to increase sample sizes and generalizability. Advanced statistical techniques, including ANCOVA, can be used to explore the influence of various factors on learning outcomes and deliverable quality. Further pedagogical refinements – such as client-developer role-playing, differentiated rubric criteria with concrete examples, and explicit training in distinguishing Compliance from Correctness – are recommended to reduce rating inconsistencies and improve measurement reliability. These efforts aim to deepen insight into the effectiveness of OIAs as a replicable framework for bridging academia-industry gaps in QA education.

7. Conclusions

This study highlights the potential of OIAs as an innovative pedagogical approach to enhance QA education in undergraduate SE. OIAs help to align academic instruction with industry practices, reducing the gap between theoretical knowledge and practical application, preparing students for real-world QA roles. The findings demonstrated that OIAs hold promise in improving group learning and project deliverable quality, particularly in metrics such as Integrity and Usability. These improvements underscore the value of integrating experiential learning techniques grounded in Vygotsky's ZPD, which fosters collaboration, reflection, and skill application.

Beyond SE, the structured approach of OIAs and their alignment with ISO/IEC 25010:2023 offer broader applicability across disciplines that emphasize quality management systems. This study contributes to QA pedagogy by providing a replicable framework that supports industry-academia collaboration while equipping students with skills essential for professional success. Future research can expand on these findings by exploring individual-level learning outcomes and implementing OIAs across diverse educational settings to further validate their effectiveness.

Acknowledgements – This work was made possible by the help of the teaching assistants for the SE class where the study was conducted. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors.

References

1. British Standards Institution, Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Product quality model, BSI, London, 2023.
2. CMMI Product Team, CMMI for Development, Version 1.3 (Technical Report CMU/SEI-2010-TR-033), Software Engineering Institute, Carnegie Mellon University, 2010.

3. R. S. Pressman and B. Maxim, *Software Engineering: A Practitioner's Approach*, 9th edn, McGraw-Hill, New York, 2019.
4. D. Suri, Software Quality Assurance for Software Engineers, *Proc. 2004 ASEE Annual Conference*, Salt Lake City, Utah, pp. 9.1106.1–9.1106.9, June 2004.
5. W. Fleischman, Electronic Voting Systems and the Therac-25: What have we learned. The Backwards, Forwards, and Sideways Changes, *ICT Proc. ETHICOMP*, pp. 170–179, 2010.
6. S. Acharya, P. A. Manohar and P. Y. Wu, Intellectual Merit and Broader Impact: Collaborative Education toward Building a Skilled Software Verification and Validation Community, *Inf. Syst. Educ. J.*, **17**(6), pp. 12–21, 2019.
7. S. Elkosantini, M. Laabidi, M. Ouertani, M. El Fekih, S. Ben Nasr, A. El Assi, K. Aissani, M. Ben Ayed, S. Mosbah, O. Ben Othman, L. Medini, M. Bourguiba and N. Lamouchi, Industrial needs v. Engineering education curricula related to maintenance, production and quality in industry 4.0: A gap analysis case study in Tunisia and Morocco, *Ind. High. Educ.*, **37**(5), pp. 634–652, 2023.
8. P. Grillo, A Comparative Investigation of Software Engineering Theory and Practice in Industry and Academia, Temple University, December 1998.
9. Y. Chen, J. Chen, Y. Gao, D. Chen and Y. Tang, Research on Software Failure Analysis and Quality Management Model, *Proc. 2018 IEEE Int. Conf. Software Quality, Reliability and Security Companion (QRS-C)*, Lisbon, pp. 94–99, July 2018.
10. T. O. A. Lehtinen, M. V. Mäntylä, J. Vanhanen, J. Itkonen and C. Lassenius, Perceived causes of software project failures – An analysis of their relationships, *Inf. Softw. Technol.*, **56**(6), pp. 623–643, 2014.
11. Y. Singh and B. Goel, A step towards software preventive maintenance, ACM SIGSOFT Softw. Eng. Notes, **32**(4), p. 10, 2007.
12. W. Y. Wong, S. Wen Yu and C. W. Too, A Systematic Approach to Software Quality Assurance: The Relationship of Project Activities within Project Life Cycle and System Development Life Cycle, *Proc. 2018 IEEE Conf. Systems, Process and Control* (ICSPC), Melaka, Malaysia, pp. 123–128, December 2018
13. H. Kaynak, The relationship between total quality management practices and their effects on firm performance, *J. Oper. Manag.*, **21**(4), pp. 405–435, 2003.
14. L. M. Leinicke, J. A. Ostrosky, W. M. Rexroad, J. R. Baker and S. Beckman, Interviewing as an Auditing Tool, *CPA Journal*, **75**(2), pp. 34–39, 2005.
15. E. Seipp and D. L. Lindberg, A guide to effective audit interviews, *CPA Journal*, **82**(4), p. 26, 2012.
16. K. Lewis, *Vygotsky's Zone of Proximal Development, in The SAGE Encyclopedia of Intellectual and Developmental Disorders*, SAGE Publications, Inc., 2018.
17. L. S. Vygotsky, *Mind in Society: Development of Higher Psychological Processes*, Harvard University Press, Cambridge, pp. 1–250, 1980.
18. S. Ashruf, D. van der Plaats, H. Al-Ers, A. Malinowska and X. Peng, Professional Practice: A Case Study of Collaborations between Education and Industry, *J. Educ. Pract.*, 2021.
19. I. Gremyr, J. Lenning, M. Elg and J. Martin, Increasing the value of quality management systems, *Int. J. Qual. Serv. Sci.*, **13**(3), pp. 381–394, 2021.
20. B. Poksinska, J. A. E. Eklund and J. J. Dahlgaard, ISO 9001:2000 in small organisations: Lost opportunities, benefits and influencing factors, *Int. J. Qual. Reliab. Manag.*, **23**(5), pp. 490–512, 2006.
21. D. H. Rhodes, R. Valerdi and G. J. Roedler, Systems engineering leading indicators for assessing program and technical effectiveness, *Syst. Eng.*, **12**(1), pp. 21–35, 2009.
22. C. Shepherd, *A Systems Engineering Approach to Quality Assurance for Aerospace Testing*, Jacobs/ESSSA Group, 2012.
23. L. Zheng, C. Baron, P. Esteban, R. Xue, Q. Zhang and S. Yang, Using Leading Indicators to Improve Project Performance Measurement, *J. Syst. Sci. Syst. Eng.*, **28**(5), pp. 529–554, 2019.
24. International Organization for Standardization, Quality management systems – Requirements (ISO 9001:2015), ISO, Geneva, 2015.
25. O. Y. Sowunmi, S. Misra, L. Fernandez-Sanz, B. Crawford and R. Soto, An empirical evaluation of software quality assurance practices and challenges in a developing country: a comparison of Nigeria and Turkey, *SpringerPlus*, **5**(1), p. 1921, 2016.
26. D. Wilson and C. S. Johnson, Education of quality and quality of education: an Australian experience, *Trans. Inf. Commun. Technol.*, **8**, 1994.
27. A. Stepanova, A. Weaver, J. Lahey, G. Alexander and T. Hammond, Hiring CS Graduates: What We Learned from Employers, *ACM Trans. Comput. Educ.*, **22**(1), pp. 1–20, 2022.
28. W. Pádua, "Using Quality Audits to Assess Software Course Projects," in *2009 22nd Conference on Software Engineering Education and Training*, Hyderabad, India: IEEE, pp. 162–165, 2009.
29. C. Y. Laporte and A. April, Software Quality Assurance in an Undergraduate Software Engineering Program, *Proc. Can. Eng. Educ. Assoc*. CEEA, June 2013.
30. M. Jenkins, Experience In Teaching Software Quality Management At The Graduate Level, *Proc. 2007 ASEE Annual Conference & Exposition*, Honolulu, Hawaii, pp. 12.711.1–12.711.12, June 2007.
31. J. B. Thompson and H. M. Edwards, How to teach practical software quality assurance. An experience report, Proc. First Asia-Pacific Conf. on Quality Software, Hong Kong, China, *IEEE Comput. Soc.*, pp. 181–187, 2000.
32. S. Acharya, P. Manohar, W. Schilling, A. Ansari and P. Wu, Collaborative Education: Building a Skilled Software Verification and Validation User Community, *Proc. 2014 ASEE Annual Conference & Exposition*, Indianapolis, Indiana, pp. 24.281.1–24.281.11, June 2014.
33. I. S. Elgrably and S. R. Bezerra Oliveira, A Quasi-Experimental Evaluation of Teaching Software Testing in Software Quality Assurance Subject during a Post-Graduate Computer Science Course, *Int. J. Emerg. Technol. Learn.*, **17**(5), pp. 57–86, 2022.
34. O. Gotel, C. Scharff and A. Wildenberg, Teaching software quality assurance by encouraging student contributions to an open source web-based system for the assessment of programming assignments, *ACM SIGCSE Bull.*, **40**(3), pp. 214–218, 2008.
35. S. Luo, J. Bian, X. Zhang and Y. Li, Research of Practical Teaching System of Software Quality Assurance and Testing Based on Large-scale Engineering Practice, *Proc. 2018 4th Int. Conf. Humanities and Social Science Research* (ICHSSR 2018), Wuxi, China, March 2018.
36. G. Moser, R. Vallon, M. Bernhart and T. Grechenig, Teaching Software Quality Assurance with Gamification and Continuous Feedback Techniques, *Proc. 2021 IEEE Global Engineering Education Conference* (EDUCON), Vienna, Austria, pp. 505–509, April 2021.

37. M. DeFond and J. Zhang, A review of archival auditing research, *J. Account. Econ.*, **58**(2–3), pp. 275–326, 2014.
38. M. Abu-Jeyyab, M. Al-Jafari, M. H. El Din Moawad, S. Alrosan and M. Al Mse'adeen, The Role of Clinical Audits in Advancing Quality and Safety in Healthcare Services: A Multiproject Analysis From a Jordanian Hospital, Cureus, **16**(2), p. e54764, 2024.
39. M. Hanskamp-Sebregts, P. Zegers, L. Mertens, A. W. D. van der Molen, J. M. T. Cappon, P. A. E. Severijnen, H. Robben, R. S. Sluijsmans, G. J. de Boer and P. W. G. Elings, Effects of patient safety auditing in hospital care: results of a mixed-method evaluation (part 1), *Int. J. Qual. Health Care*, **31**(7), pp. 8–15, 2019.
40. M. M. Dolcini, G. J. Gandelman, J. V. Foster, S. E. Dorsey and N. P. Reitz, Use of effective training and quality assurance strategies is associated with high-fidelity EBI implementation in practice settings: a case analysis, *Transl. Behav. Med.*, **11**(1), pp. 34–45, 2021.
41. W. R. Knechel, G. V. Krishnan, M. Pevzner, L. B. Shefchik and U. K. Velury, Audit Quality: Insights from the Academic Literature, *Audit. J. Pract. Theory*, **32**(S1), pp. 385–421, 2013.
42. R. De Kleijn and A. Van Leeuwen, Reflections and Review on the Audit Procedure: Guidelines for More Transparency, *Int. J. Qual. Methods*, **17**(1), p. 1609406918763214, 2018.
43. K. Watson, Utilization of active and cooperative learning in EE courses: three classes and the results, *Proc. 1995 Frontiers in Education Annual Conf.*, Atlanta, GA, pp. 3c2.1–3c2.6, October 1995.
44. S. Anwar, M. Menekse and A. Kardgar, Engineering Students' Self-Reflections, Teamwork Behaviors, and Academic Performance, *Proc. 2019 ASEE Annual Conference & Exposition*, Tampa, Florida, p. 32738, June 2019.
45. R. S. Matthews, J. L. Cooper, N. Davidson and P. Hawkes, Building Bridges Between Cooperative and Collaborative Learning, *Change Mag. High. Learn.*, **27**(4), pp. 35–40, 1995.
46. K. Cho and C. MacArthur, Learning by reviewing, *J. Educ. Psychol.*, **103**(1), pp. 73–84, 2011.
47. R. M. García-García, C. A. García-Caballero, R. Villarreal-Lara and V. A. Mirón-Mérida, On-site inspection as a tool for developing auditing skills in engineering: integrating continuous education contents into higher education programs, *Front. Educ.*, **9**, p. 1416110, 2024.
48. N. Fila, T. Fernandez, S. Purzer and A. Bohlin, Innovation and the Zone of Proximal Development in Engineering Education, *Proc. 2016 ASEE Annual Conference & Exposition*, New Orleans, Louisiana, p. 27312, June 2016.
49. S. McLeod, Vygotsky's Zone of Proximal Development and Scaffolding, Simple Psychology, November 2023, Accessed 08 August 2025.
50. J. W. Creswell and T. C. Guetterman, *Educational Research: Planning, Conducting, and Evaluating Quantitative and Qualitative Research*, 6th edn, Pearson, New York, 2019.
51. J. W. Creswell and J. D. Creswell, *Research design: Qualitative, quantitative, and mixed methods approaches*, 5th edn, SAGE Publications, Inc., Thousand Oaks, 2018.
52. F. N. Che, K. D. Strang and N. R. Vajjhala, Using Experiential Learning to Improve Student Attitude and Learning Quality in Software Engineering Education, *Int. J. Innov. Teach. Learn. High. Educ.*, **2**(1), 2021.
53. W. N. Kernan, C. M. Viscoli, R. W. Makuch, L. M. Brass and R. I. Horwitz, Stratified Randomization for Clinical Trials, *J. Clin. Epidemiol.*, **52**(1), pp. 19–26, 1999.
54. S. D. Gantz, *IT Audit Processes, in The Basics of IT Audit*, Elsevier, New York, pp. 149–165, 2014.
55. N. Hill, J. Brierley and R. MacDougall, *How to Measure Customer Satisfaction*, 2nd edn, Routledge, Oxford, 2017.
56. P. C. Wade, Measuring the Impact of Oral Inquiry Audits on Student Outcomes and Software Quality: A Pedagogical Innovation in Software Engineering Education, Ph.D. thesis, Texas A&M University, College Station, Texas, 2025 [unpublished].
57. L. Ott and M. Longnecker, An introduction to statistical methods and data analysis, 7th edn, Cengage Learning, Boston, 2016.
58. A. Kolmos, and J. E. Holgaard,Responses to Problem Based and Project Organised Learning from Industry, *International Journal of Engineering Education*, **26**(3), pp. 573–583, 2010.
59. A. Drobnič Vidic, The impact of problem-based learning on statistical thinking of engineering students, *International Journal of Engineering Education*, **27**(4), pp. 885–896, 2010.

Appendix A: Glossary

ANCOVA	Analysis of covariance	QA	Quality assurance
CMMI	Capability maturity model integration	QC	Quality control
CSCE	Computer Science and Engineering	SDLC	Software development lifecycle
EL	Experiential Learning	SE	Software engineering
GPA	Grade point average	SQA	Software quality assurance
ICT	Information and communication technology	SV&V	Software verification and validation
ISO	International Organization for Standardization	ZPD	Zone of proximal development
OIA	Oral inquiry audit		

Pauline Wade, at the time of writing, is researching impact of OIAs in an SE class while pursuing a doctorate degree at Texas A&M University, Department of Multi-Disciplinary Engineering. She is concurrently serving as professor of practice in the Department of Computer Science & Engineering

Hillary E. Merzdorf holds a PhD in engineering education from Purdue University. Her research interests are in digital assessment for STEM education, learning design and educational technology, and spatial cognition with representations in engineering. Her background is in educational psychology and human factors engineering.

Tracy Hammond is a professor in the Department of Computer Science & Engineering. Her research interests include: sketch recognition, perception, cognitive behavior, computer human interaction, artificial intelligence, concept learning, computer graphics, psychology, anthropology, and the gender gap in computer science.