

Research Article

Cite this article: Annamalai, N., & Nasor, M. (2025). Insights into Computational Thinking E-Assessment Tool (Cte-AT) In Engineering Education: A Preliminary Study of Student and Educator Perspectives. *Educational Process: International Journal*, 16, e2025269. <https://doi.org/10.22521/edupij.2025.16.269>

Received February 27, 2025

Accepted May 6, 2025

Published Online June 21, 2025

Keywords: Computational thinking, e-assessment, engineering students, higher education

Author for correspondence:

Mohamed Nasor

 m.nasor@ajman.ac.ae

 Department of Biomedical Engineering, College of Engineering and Information Technology, Ajman University, Ajman, UAE

**OPEN ACCESS**

© The Author(s), 2025. This is an Open Access article, distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>), which permits unrestricted re-use, distribution, and reproduction, provided the original article is properly cited.

Insights into Computational Thinking E-Assessment Tool (Cte-AT) In Engineering Education: A Preliminary Study of Student and Educator Perspectives

Nagaletchimee Annamalai , Mohamed Nasor 

Abstract

Background/purpose. This study presents the findings of an initial investigation into developing a Computational Thinking-Enhanced Assessment Tool (CTe-AT) based on computational thinking principles.

Materials/methods. We conducted comprehensive interviews with 10 lecturers and another 10 students to gain a deeper insight into educators' and students' viewpoints. The analysis of these interviews involved a thematic approach, utilising the steps outlined by Braun and Clarke (2006) to identify and categorise the emerging themes.

Results. The participants acknowledged the anticipated advantages of the CTe-AT. Educators and students foresee enhancements in computational abilities, individualised feedback provision, and facilitation of the shift toward hybrid learning. The objective was to ensure engineering students not only grasp theoretical principles but also cultivate practical skills essential for success in the dynamically changing realm of engineering. Emphasis on critical and creative thinking and potential benefits for independent learning and career preparation strengthens the case for intentional efforts to integrate computational thinking into the educational framework.

Conclusion. The study addresses the need for human evaluation, particularly in assessing presentation and collaboration skills. Educators and students emphasise the nuanced nature of these skills, suggesting a reliance on human judgment for accurate assessment. In conclusion, the positive response from educators and students provides a solid foundation for the practical design and implementation of the CTe-AT.

1. Introduction

Throughout the years of experience in higher education institutions, the researchers in this study have observed that assessment plays a crucial role in shaping students' learning journeys. How students are assessed influences their motivation and determines their strategies to achieve academic success. Many students view assessment as a pivotal factor in their educational experience, guiding their learning efforts and shaping their future opportunities. Therefore, assessing students' accomplishments is a crucial aspect of formal education, serving as the record for academic degrees. It also plays a significant role in pinpointing students' Zone of Proximal Development (ZPD) and guiding their continued participation in learning activities (Falkner & Sheard, 2019).

A critical component of the assessment is understanding the methods by which instructors evaluate students' work and providing the students with similar knowledge of the assessment process. Students become more engaged in learning by comprehending the criteria and methods employed to evaluate their performance. This transparency encourages self-regulation, allowing students to critically evaluate their progress and assume responsibility for their academic development. Moreover, providing students with this knowledge fosters a more constructive and collaborative relationship between instructors and learners, improving the learning environment. Nevertheless, the process of student assessment presents a significant challenge. According to Grover & Pea (2013), to conduct a proper assessment, it is essential to have a validated measurement instrument.

Accordingly, this study examined engineering students' and teachers' perceptions of developing a software termed CTe-T. This investigation underscores the significance of developing an evaluation software that prioritises computational thinking for engineering students' final projects. Engineering disciplines are continuously advancing due to technological breakthroughs. To keep up with these advancements, engineering students must understand theoretical concepts and develop practical skills (Boshradi & Hosseini, 2023). Engineers today face numerous complex, poorly defined, and multi-dimensional challenges, and the knowledge needed to solve them is typically spread among various individuals.

In response to these challenges, this current study examined engineering students' and teachers' perceptions of developing a software tool called CTe-T. This investigation underscores the need for innovative evaluation software that integrates computational thinking into the assessment of final-year engineering projects. Traditional assessments often emphasise technical proficiency without fully capturing students' problem-solving, critical thinking abilities, and important skills required in today's rapidly evolving engineering landscape.

Computational thinking (CT) is a highly relevant concept for engineering projects and real-world applications. It offers a systematic framework for problem-solving by deconstructing complex problems into manageable components and addressing them through logical and algorithmic reasoning (Tsai et al., 2020; Rojas-Lopez & García-Penalvo, 2019). CTe-T seeks to provide a structured and objective approach to evaluating learners' abilities beyond traditional methods by embedding computational thinking into assessment.

This instrument is necessary to assess students based on their technical proficiency and capacity to implement computational thinking principles. For the complete implementation of CT, it is necessary to evaluate the competencies that students acquire (Varela et al., 2019). Consequently, the knowledge of assessment will offer students a deeper understanding of their ability to apply these abilities to intricate, real-world engineering challenges.

CT encompasses a diverse set of mental ideas and resources derived from computer science, which aid students in analysing and developing solutions to issues within their respective fields. These

include methods such as breaking down problems, abstraction, algorithms, data representation and analysis, and automation (Roman et al., 2017). Acquiring computational thinking skills starts with grasping the concept of abstraction as a fundamental tool for logical thinking, and it offers a range of educational advantages that enhance cognitive abilities and comprehension of various subjects (Varela et al., 2019).

CT involves a structured approach to problem-solving using computational methods and is increasingly acknowledged as a crucial skill for engineering graduates. Furthermore, an assessment that integrates computational thinking prepares engineers to transform into innovative entrepreneurs, which is pertinent as it empowers effective participation in global markets (Barak & Usher, 2019). The significance of fostering an innovative mindset underscores the need to cultivate and evaluate this capability in engineering students who are anticipated to join the competitive and ever-evolving job market (Barak & Usher, 2019). New educational initiatives call for an updated conceptualisation of innovative thinking and comprehensive assessment tools (Usher & Barak, 2018). Using CTe-T, engineering educators can enhance students' computational thinking abilities while ensuring fair and standardised assessment practices.

Developing CTe-AT for engineering students may mitigate potential bias during the evaluation process. Despite educators' best intentions, they can inadvertently introduce bias into their assessments, which may affect students' learning outcomes. Using a standardised assessment tool, educators and students can establish a fair and consistent framework that objectively evaluates students' performance based on predefined criteria, reducing the likelihood of subjective biases. This ensures that all engineering students receive an unbiased evaluation, fostering a learning environment where everyone has an equal opportunity to succeed and grow in their computational thinking skills.

1.1. The Computational Thinking E-Assessment Software (CTe-AT)

A comprehensive rubric will be developed, aligning with the fundamental principles of computational thinking, including problem decomposition, pattern recognition, abstraction, algorithm design, and algorithm evaluation. This rubric will guide assessors in appraising students' performance in their final-year projects. The e-assessment tool will comprise the following components:

I Evaluation Using a Rubric

Table 1. Evaluation Using a Rubric

Definition	Assessment Focus	Exemplary Performance	Needs Improvement
Decomposition The ability of students to break down complex engineering problems into smaller and manageable components.	Recognised the key sub-problems and their relationships. Establish a structured approach to solving the problem. Effectively organises tasks to support project execution.	Clearly and systematically analyse the problems into logical, interdependent sub-problems with justification.	Problem breakdown is unclear, lacks structure, vague or overlooks important components, leading to an incomplete solution.
Pattern of Recognition The ability to identify recurring themes,	Recognises similarities with existing problems or solutions.	Demonstrates a keen ability to detect patterns and leverage prior	Fails to recognise relevant patterns, leading to redundant

trends, or structures within the problem to develop efficient solutions.	Applies past knowledge and industry best practices to inform decisions. Draws connections between different aspects of the project.	solutions to optimise the project approach.	work or inefficient problem-solving.
Abstraction The ability to focus on important details while filtering out unnecessary information to simplify problem-solving.	Determine key variables and constraints impacting the project. Develops generalised models or frameworks for broader application Balances detail and complexity to improve clarity and efficiency.	Demonstrates strong abstraction skills, effectively simplifying complex issues while preserving critical functionalities.	Struggles to distinguish important details, leading to either oversimplified or overly complex solutions.
Algorithm The ability to create a step-by-step plan or logical sequence of actions to solve the problem efficiently.	Develops step-by-step and logical procedures. Ensures clarity, efficiency, and feasibility in solution design. Considers edge cases and potential failures.	Designs a robust, efficient, and scalable algorithm with clear implementation steps and contingency plans.	The proposed algorithm lacks structure and efficiency, or fails to address key problem aspects.

II. Integration into Project Evaluation

The e-assessment of computational thinking will become integral to the year final-year project evaluation process. It will be scored alongside traditional engineering project criteria, ensuring a holistic evaluation of students.

III. Faculty Development

Faculty members will undergo training and receive resources to assess computational thinking skills effectively in engineering projects. This will ensure consistent and fair evaluation across the institution.

IV. Implementation Plan

The implementation of this assessment tool will be carried out in stages, starting with faculty training and the development of assessment rubrics. Subsequently, pilot testing will be conducted, allowing for iterative improvements based on feedback. A full-scale implementation will be initiated in subsequent academic years.

The ADDIE model is a framework for instructional design that is frequently employed and offers a structured, systematic approach to developing effective educational experiences (Shakeel et al., 2023). The ADDIE model was considered in this study for its systematic and iterative nature, which allows a structured development process that can be continuously refined based on feedback (Lee et al., 2017). Compared to other instructional design models, such as the Dick and Carey Model (1996)

or the Successive Approximation Model (Raaijmakers & Shiffrin, 1981). The ADDIE model permits a clear, linear progression while allowing for flexibility through formative evaluation at each phase. This is particularly pertinent in developing an e-assessment tool, as it allows ongoing adjustment with engineering education needs.

Furthermore, the ADDIE model provides a more straightforward, more appropriate framework for structured educational settings, such as higher education institutions. Also, it emphasises evaluation at multiple levels, which ensures that the computational thinking assessment tool is continuously improved to increase its effectiveness and reliability. By considering the ADDIE model, the current study ensures that the development of the computational thinking e-assessment tool follows a rigorous, evidence-based instructional design process, ultimately leading to a more effective and scalable assessment approach.

It is an acronym that denotes the five phases of the process: Analysis, Design, Development, Implementation, and Evaluation.

- Analysis: The initial phase of the ADDIE model involves identifying and understanding the learning objectives, goals, and requirements. During this phase, instructional designers and educators gather critical information to establish the project's scope and problem.

- Design: In the Design phase, the instructional strategy is developed based on the data collected during the analysis. This entails specifying the course structure, content, assessments, delivery methods and instructional strategies.

- Development: This is the actual creation of instructional materials. It is where the concepts and strategies outlined in the design phase are translated into tangible learning resources.

- Implementation: The materials that have been developed are distributed to learners during the Implementation phase.

- Evaluation: The ADDIE model is characterised by a continuous evaluation process across all phases. The evaluation is classified into two categories: Formative Evaluation (continuous, performed during each phase) and Summative Evaluation (final, conducted after implementation). This phase is designed to evaluate the efficiency and efficacy of the instructional design process and its results.

This study reported on the analysis phases, which include students' and teachers' perceptions of developing the CTe-AT. In this paper, we present the preliminary results of a study that explored educators' and students' perspectives regarding the creation of such assessment software. This initial inquiry is pivotal in laying the groundwork for our overarching objective, illuminating the crucial factors, challenges, and prospects that will inform the software's design and implementation. The insights garnered from this research will substantially contribute to refining and optimising the forthcoming software, ensuring it effectively aligns with the requirements and expectations of educators in engineering education. The research question for this study is: What are educators' perceptions of the development of software tools (CTe-AT for assessing engineering students)?

2. Literature Review

The following section delves into the theory and framework that directed this research.

2.1. Social Constructivism

Social constructivism was initiated by Vygotsky (1978) as an educational philosophy that underscores the proactive role of learners in acquiring knowledge. According to this theory, learning is far from passively absorbing information. It is an engaged process where individuals construct their comprehension by building upon their past experiences, beliefs, and cognitive frameworks (Vygotsky, 1978). Utilising the tenets of social constructivism in developing an evaluation instrument for

engineering students enables educators to go beyond mere knowledge measurement. These assessments also nurture active learning, problem-solving abilities, teamwork, and an enduring commitment to continuous learning - all of which are pivotal for achieving success in engineering. Constructivism posits that the most effective learning is initiated by activating and harnessing this prior knowledge. Hence, assessments should begin by acknowledging and utilising what students already know about engineering and computational thinking.

Constructivism also champions the idea that learners best construct knowledge when they actively grapple with problems and apply their learning to real-life situations. This approach fosters active learning as students tackle genuine, intricate problems that mimic scenarios they may confront in their forthcoming professional journeys.

2.2. Integration of Bloom's Taxonomy, Computational Thinking, and 21st-century skills

Bloom Taxonomy

The study was also guided by Bloom's taxonomy, a widely used educational framework that has been instrumental in discussions on effective teaching for more than five decades (Athanassiou et al., 2003). In the revised version of Bloom's taxonomy by Anderson and Krathwohl (2001), synthesis switched places with evaluation and was renamed "create" to promote broader teacher usage, more overlap, and a less rigid hierarchy.

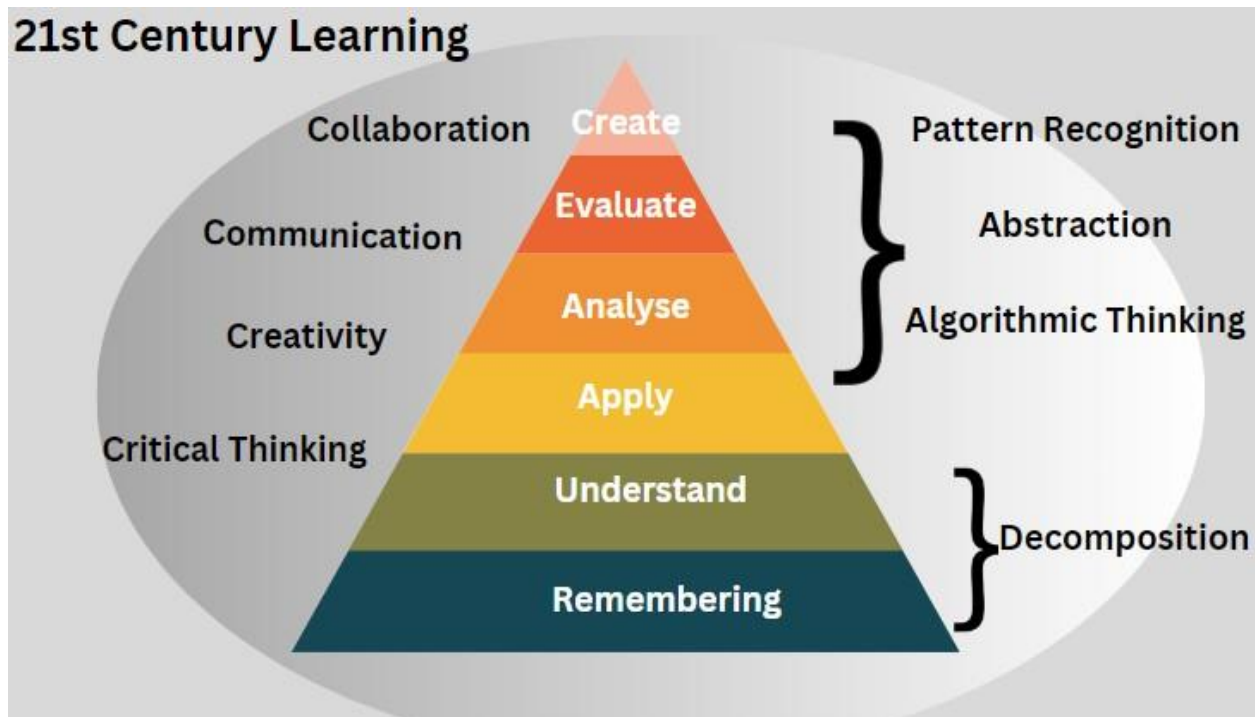
In designing CTe-AT, integrating computational thinking within Bloom's Taxonomy encourages problem-solving and innovation. Educators can design curricula and assessments that encourage computational thinking at all levels. The practical integration of computational thinking into Bloom's Taxonomy can be achieved through project-based learning, coding exercises, and digital problem-solving activities. Cross-disciplinary projects can emphasise the real-world applications of computational thinking in science, engineering, and beyond.

The 21st-century learning emphasises critical thinking, creativity, collaboration, and communication. In fields such as engineering, these abilities are essential for achieving success in a swiftly evolving and technology-driven world. Student learning outcomes can be improved by incorporating these components into educational frameworks, mainly through the lens of Social Constructivism and Bloom's Taxonomy. Social constructivist classrooms emphasise cooperation and problem-solving, which are reflective of the 21st-century skills related to communication and collaboration. Applying Bloom's Taxonomy with 21st-century learning principles facilitates a structured approach to developing these skills.

The key components of computational thinking, shown in a pyramid in Figure 1, include:

- **Decomposition:** involves breaking down complex problems into smaller, more manageable sub-problems or tasks. It simplifies the problem-solving process by focusing on individual elements rather than tackling the entire problem simultaneously.
- **Pattern Recognition:** Recognising patterns and trends within data or information is critical to computational thinking. It helps identify similarities and regularities that can be used to solve problems more efficiently.
- **Abstraction:** Abstraction involves simplifying complex reality by modelling the problem at a high level of detail while ignoring irrelevant details. It helps create a simplified problem representation.
- **Algorithmic Thinking:** This involves developing a set of step-by-step instructions or algorithms to solve a specific problem or perform a task. Algorithms are essential in computational thinking as they provide a structured approach to problem-solving.

Figure 1. The Relationship of 21st-century Learning, Bloom's Taxonomy and Computational Thinking



3. Methodology

This study utilised a case study approach to examine educators' and students' perceptions of an AI-based assessment tool in an engineering course, situating the research within the interpretive paradigm. Adopting an interpretive perspective in a case study enables researchers to explore participants' intricate and subjective experiences within a particular educational context (Patton, 2002).

3.1. Research Design

This study adopted a qualitative case study research design to assess computational thinking skills in engineering training programs in depth. This study aimed to delve deeply into educators' and students' perceptions of CTe-AT to assess engineering students' comprehension of it thoroughly, rather than test hypotheses.

The qualitative research approach is chosen for its capacity to allow for a detailed exploration of educators' and students' perceptions of CTe-AT. The primary method for gathering data was semi-structured interviews, chosen for their ability to extract context-specific and information-rich insights, shedding light on the multifaceted nature of educators' and students' viewpoints.

3.2. Participants

The students were recruited from a university's Information Communication and Technology and Engineering Department in the United Arab Emirates. The department is responsible for developing the CTe-AT. Purposive sampling was employed to identify educators proficient in computational thinking and actively engaged in instructing engineering students and administering assessments. Similarly, purposive sampling was implemented to select the student participants pursuing degrees in engineering and technology. These students have a strong foundation in computational thinking and related skills. The subject matter is well-known to them, which guarantees that they offer constructive feedback and valuable insights.

Thirty students and 15 educators expressed interest, but only ten educators and ten students were selected based on data saturation. However, upon conducting a preliminary analysis, it was

determined that data saturation was obtained with 10 educators and 13 students. The data saturation took place when additional data no longer introduced new ideas, themes, perspectives, or variations, ensuring that further interviews would be redundant and would not significantly contribute to the study's findings (Howitt, 2012). However, the data collected from the remaining participants was not ignored entirely. Instead, it was reviewed and compared with the selected data to confirm saturation and ensure no significant insights were overlooked. These additional data points reinforced the robustness of the study's findings by validating recurring themes, thereby enhancing the credibility of the research.

The primary goal is to understand how educators and student participants perceive the design and utilisation of assessment tools related to computational thinking. Before starting each interview, participants were provided with a thorough explanation of the study's objectives. The students were presented with a consent form that outlined the study's goals and the potential advantages of their involvement, explicitly stating the absence of any associated risks.

Ethical considerations are of utmost importance in this study. Informed consent was obtained from all participants, ensuring their rights and privacy were upheld. Confidentiality was preserved by assigning pseudonyms to participants, and all data will be securely stored and accessible only to the research team. In presenting the findings, we utilised pseudonyms for the participants. The educator participants are labelled as P1, P2, P3 and student participant are labelled as PA1, PA2, PA3...

Table 2. Demographic Information

Educator Participant	Age	Course Taught
P1	55	Engineering
P2	43	Engineering
P3	45	Engineering
P4	43	Engineering
P5	56	Information Technology
P6	47	Information Technology
P7	44	Information Technology
P8	53	Information Technology
P9	47	Information Technology
P10	48	Information Technology
Student Participant	Age	Major
PA1	20	Engineering
PA2	21	Engineering
PA3	20	Engineering
PA4	20	Engineering
PA5	19	Engineering
PA6	22	Information Technology
PA7	22	Information Technology
PA8	20	Information Technology
PA9	23	Information Technology
PA10	22	Information Technology

4. Data Collection

The principal data collection method was semi-structured face-to-face interviews. The interview protocol was developed to explore participants' experiences, opinions, and recommendations for designing and applying assessment tools centred on computational thinking. Semi-structured interviews facilitated a thorough exploration of participants' perspectives, encouraging them to articulate their thoughts, challenges, and innovative ideas in their own words. Utilising interviews as the primary data collection method enables an investigation of context-specific aspects of educators' and students' viewpoints. The interviews included probe-type questions designed to encourage more elaboration and clarify responses. For example, these questions focused on perception and challenges in assessment design, the impact of computational thinking tools, and participants' recommendations for future improvements. The following are the research questions:

1. What are your initial impressions of the Computational Thinking-Enhanced Assessment Tool (CTe-AT)?

- What aspects of the tool do you find most valuable? Can you detail your answers?
- Do you foresee any challenges in its implementation? Can you explain with examples?

2. How do you think CTe-AT can enhance students' computational thinking abilities?

- Can you provide examples of how computational thinking is currently integrated into your teaching?
- How might this tool improve students' problem-solving skills?

The researcher's role in this study was to act as a facilitator, guiding the interviews while ensuring that participants felt comfortable sharing their perspectives. The researcher maintained neutrality throughout the process, encouraging open responses while minimising bias.

4.1. Data Analysis

Thematic analysis, a widely recognised qualitative research approach, was employed to analyse the interview data. This methodology permits the identification of recurring patterns, themes, and narratives within the collected data. An iterative coding and categorisation process distilled educators' shared understandings, contrasting viewpoints, and emerging insights concerning the design and use of assessment tools for computational thinking. Following the principles of qualitative analysis, this process necessitated the initial classification of participant views into codes, the systematic organisation of these codes into categories, and the final integration of these categories into sub-themes or themes. This structured approach ensured a holistic and rigorous analysis of the dataset.

Braun and Clarke's (2006) five-step thematic analysis guided the interview data analysis.

– Familiarisation: Step 1 involved coders thoroughly immersing themselves in the data by reviewing and rereading the transcripts line by line. This stage required a comprehensive engagement with the material to understand it deeply.

– Initial codes: In Step 2, the coders created initial codes to highlight important content within the transcripts, annotating as needed. This step also demanded a comprehensive data review, scrutinising each line to ensure a profound comprehension of the material.

– Identifying themes: Step 3 involved a different research assistant examining all annotated transcripts and addressing any discrepancies in interpretation. The authors resolved these discrepancies and used the codes to identify categories. During this phase, recurring patterns of significance within the dataset were identified as themes.

– Reviewing themes: In Step 4, the research team considered combining specific themes or breaking them down into subthemes, iterating this process until the thematic map met their expectations. Researchers assessed the themes to determine if they needed further refinement, consolidation, or subdivision into subthemes.

– Defining and naming the themes: In Step 5, the research team defined and named the themes, choosing names that effectively conveyed their significance. They provided precise definitions for each theme to ensure effective communication of their importance.

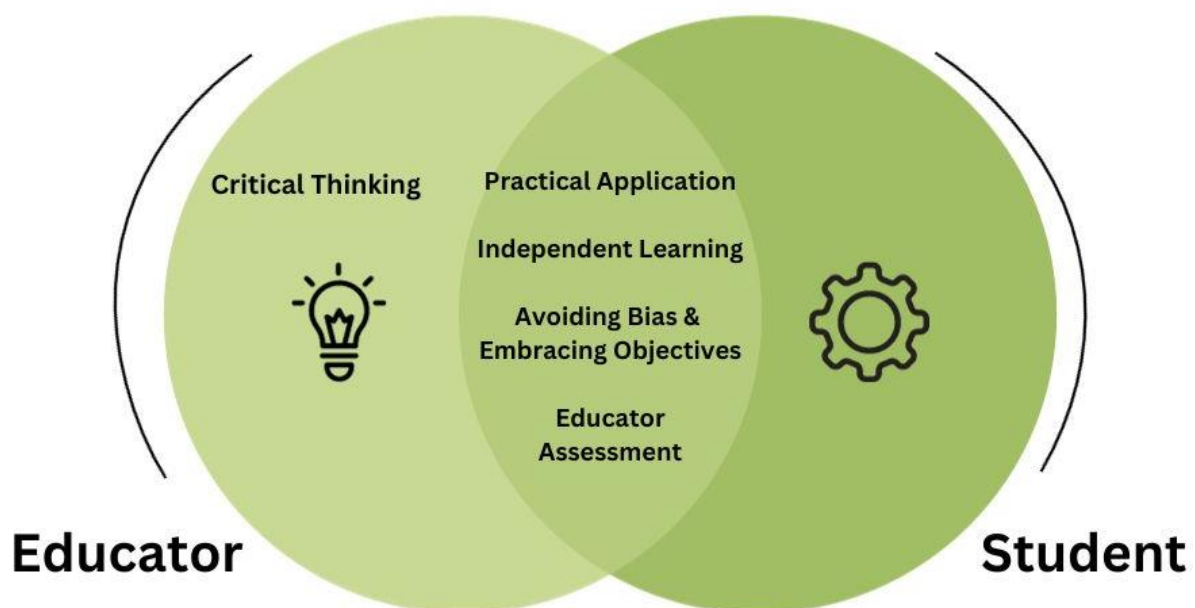
4.2. Trustworthiness of the Qualitative Data

Transferability was confirmed by providing a detailed description of the participants and the study settings. The research procedures and data collection tools were also thoroughly explained so that the findings apply to other contexts. In this research, member checking was conducted by having participants review the interpretation of data and offer their input. Any instances of data misinterpretation, as pointed out by the participants, resulted in the removal of the corresponding ideas or sentences, as suggested by Maxwell (2005). Two coders independently coded the data, and the consistency ratio was calculated. To determine inter-rater reliability, we followed the method recommended by Miles and Huberman (1994). The inter-rater reliability was assessed using a raw percentage, which was in line with the guidance of Miles and Huberman (1994). According to Miles and Huberman, for the findings to be considered valid, a consensus of at least 70% among the coders is necessary. The agreement reached was 85%. Additionally, the interview schedule underwent a pretest, enhancing the quality of the interview questions. The validity and appropriateness of the questions were verified through testing with educators who were not part of the study.

5. Findings

After acquiring the data, the educators' findings were further compared with the information obtained from the students. The following findings delve into the various themes that emerge from data analysis related to educator and student perceptions of CTe-AT. Figure 2 illustrates the themes that emerged from the interview data. These themes are described in more detail in the following section.

Figure 2. Themes Related to CTe-AT



5.1. Practical Application

Participants' viewpoints shared in interviews highlight a perceived gap between theoretical knowledge and practical application in engineering education, particularly in computational skills. The educators argued that while computational courses are often part of the engineering curriculum, there is a need for a dedicated assessment tool to bridge the gap and evaluate students' ability to apply computational thinking in their final-year projects. P1 said, *"The nature of study for engineering courses should consider computational skills to develop innovations."* One of the participants raised a point about the current assessment methods (traditional assessment); some components are related to computational thinking, and some need clearer and more comprehensive definitions. This suggests a need for more precision in defining what constitutes computational thinking in the context of engineering education. In P7's words, *"in the current assessment, such components are there, but probably such a definition of computational thinking should include operational definitions."* Similarly, another participant said, *"It is a good idea, but not all the computational thinking components are included. Only decomposition and algorithmic thinking are possible. Therefore, a bit of alteration is needed."* (P9). P6 felt that assessment work can be *"easy, effective, and more automated with CTe-AT."* The proposal for CTe-AT implies a desire to formalise the evaluation of computational thinking skills. Such a tool could provide a structured framework for assessing how well students can apply computational concepts and problem-solving approaches in real-world engineering projects. This aligns with the idea that practical application, especially in the form of final-year projects, serves as a meaningful and tangible way to gauge the effectiveness of students' computational skills.

In summary, the excerpts advocate for a more intentional integration of computational skills into engineering education and suggest that a dedicated assessment tool with clear definitions could help bridge the gap between theory and practice. This approach ensures that students learn theoretical concepts and develop the practical skills necessary for success in the rapidly evolving engineering field.

Educators felt that the CTe-AT should be given to the students. P6 thought that it was a practical approach since:

There are many courses. I don't have time, so such software will allow students to improve based on feedback. Students will be aware of their strengths and weaknesses.

Student participants expressed a positive perspective regarding introducing CTe-ATs into their learning environment. Participant PA1 articulated that the software holds promise in assisting with project work, offering a potential reduction in complexity. PA1 said, *"It is a good idea since it will help us with our project work and reduce complexity."* This sentiment resonated with Participant PA2, who emphasised the helpfulness of clearly outlined steps in the learning process. PA3 stated, *"When steps are clearly outlined, it will be helpful."* PA7 highlighted the relevance of CTe-AT in transitioning to hybrid learning. PA7 emphasised the software's significance in mitigating the need for exclusive reliance on face-to-face interactions to understand project assessments. The following excerpt highlights her idea:

Since we are moving to hybrid learning, it is better to have such software and not rely on face-to-face interaction to understand how our projects are assessed.

5.2. Critical Thinking

A common theme highlighted by educators is the acknowledgement that the modern job market requires engineers who can think critically, analyse data, and devise innovative solutions using computational methods. The notion that graduates possessing robust computational thinking skills are more likely to thrive in their careers and make valuable contributions to the workforce stands out

as a key observation. P5 opined that *“the stages in computational thinking refer to critical and creative thinking and should be an important component in assessment. Such thinking is necessary for students to be successful in the future workplace.”*

The idea that providing students with tools to encourage independence in their work can improve critical thinking aligns with the overarching goal of preparing students for the challenges they will encounter in their professional journeys. P7 emphasised that *“if CTe-AT is given to students, they can work on it independently and become more critical.”* In a similar vein, P15 felt that:

...the efforts in this direction would be interesting and worthwhile for the students.
Students can also use it for their convenience and prepare for the exam. They will be more prepared for career advancement.

The emphasis on critical and creative thinking, along with the potential benefits for independent learning and exam preparation, enhances the depth of the educator participants' support for efforts to integrate computational thinking into the educational framework. There were no comments from the student participants.

5.3. Independent Learning

The concept that students can use computational tools for convenience and exam preparation further underscores the practical advantages of integrating computational thinking into education. P12 opined that CTe-AT allows students to *“indulge in self-study.”* P11 felt that *“it should be a practical software for students to guide them with their assessment.”* Therefore, they know their weaknesses and will improve *“their work and knowledge.”* P13 said that *“it is a more practical software that allows students to fully understand the concepts that they have learned.”*

Student participants also believed that the CTe-AT would facilitate self-directed learning. CTe-AT not only defines the components of the assessment but also lays out a set of rubrics that guide their learning. For example, PA6 highlighted that *“we have instructions and guidelines so we can work independently.”* Similarly, PA5 said, *“It specifies each component and rubrics, and we do not need the lecturer to detail how our projects are addressed.”*

CTe-AT acts as a silent mentor, ensuring that the concepts imparted in the classroom are grasped and deeply understood. It becomes the guardian of our independent exploration, a tool that validates students' comprehension and sharpens their skills. As students engage with the software, it becomes clear that it is more than just a medium; it is the enabler of our autonomy.

5.4. Avoiding Bias and Embracing Objectivity

Several participants felt that using an assessment tool based on computational thinking reduces bias and fosters objectivity during the evaluation. For example, P2 stated that *“educators tend to have favouritism and like certain students, and their assessments tend to be biased. Therefore, software to assess the student's assignment and project work will be an advantage.”* Another participant felt that the outlined levels for each criterion would benefit the assessment. Similarly, P7 asserted that *“create standardised rubrics that offer distinct guidelines for evaluating various facets of the project.”*

Student participants were similarly convinced that using steps and rubrics implies a structured and transparent process. Having predefined steps ensures that evaluators follow a consistent and objective path, making it less likely for subjective biases to influence the outcome. PA4 said that *“steps are set, and there are rubrics in the software; therefore, there should not be any bias.”* PA9 asserts that *“every phase is clearly outlined. Therefore, we can avoid biases. The software captures the lecturer's demands, and we can use it as a guideline.”* This confidence is likely derived from the belief that the precise steps and rubrics safeguard against subjective judgments or favouritism.

Another participant opined that *“it avoids confusion and since the machine is evaluating, there are no biases”* (PA9). Referring to the software as a guideline implies that it serves as a reference point, ensuring that assessments are based on objective criteria rather than subjective interpretations. This aligns with the idea of using predefined rules to guide the evaluation process.

5.5. Educator Assessment

The educators expressed that certain aspects of assessment, specifically presentation skills and collaboration among students, require human evaluation. P7 said that *“assessment involves presentation skills and collaboration among students, and this can only be assessed by the educators and not a machine.”* This suggests that machines may not possess the nuanced understanding or qualitative judgment needed for assessing these interpersonal and communication skills. This is because *“educators’ assessment is important as they understand their students’ work and performance much better.”*

Similarly, Students felt that lecturers are uniquely positioned to evaluate students due to their more profound understanding of individual students. This perspective implies that lecturers, through continuous interaction and observation, can grasp their students' strengths, weaknesses, and learning styles. For example, PA5 said, *“Lecturers should evaluate since they know their students better and not a software.”* One of the participants emphasised that lecturers are likely to be more tolerant and understanding when assessing their students. This could be attributed to their personal rapport and insights into each student's circumstances, challenges, and efforts. PA7 clarified that *“they will be more tolerant and understand their students’ performance.”* PA9 asserted that *“educators will understand the handwork [effort] put forward by their students.”*

6. Discussion

This aligns with the broader trend in industries increasingly relying on computational approaches for problem-solving and decision-making. The initial investigation into developing a CTe-AT based on computational thinking has produced positive outcomes, garnering favourable feedback from educators and students. The inventive approach to incorporating computational thinking into the assessment process has been well-received by educators, who recognise the capacity to nurture crucial problem-solving skills in students. Similarly, students have responded positively to developing CTe-AT, acknowledging the engaging and thought-provoking nature of the e-assessment format. This favourable reception highlights the feasibility of the designed e-assessment and indicates its potential to improve the learning experience by fostering a more profound understanding of computational thinking concepts. As the study advances, these initial positive impressions lay a robust groundwork for further exploration and advancement in technology-enhanced assessments.

The proposal for CTe-AT implies a desire to formalise the evaluation of computational thinking skills. Such a tool could provide a structured framework for assessing how well students can apply computational concepts and problem-solving approaches in real-world engineering projects. This aligns with the idea that practical application, especially in the form of final-year projects, serves as a meaningful and tangible way to gauge the effectiveness of students' computational skills.

Participants’ perceptions bridge the theoretical gap between Bloom’s Taxonomy and computational thinking. It encourages the development of creative problem solvers capable of addressing complex challenges. The future of education lies in recognising the interconnectedness of these theories and their potential for shaping well-rounded, digitally literate individuals.

Educators and students acknowledge the anticipated advantages of this tool, such as enhancing computational abilities, providing individualised feedback, and easing the shift towards hybrid learning. The objective is to guarantee that engineering students not only comprehend theoretical principles but also cultivate the practical skills essential for success in the dynamically changing realm

of engineering. The emphasis on critical and creative thinking and the potential benefits for independent learning and career preparation strengthen the case for intentional efforts to integrate computational thinking into the educational framework. These findings underscore the significance of equipping engineering students with a well-rounded skill set that extends beyond technical proficiency, aligning with the dynamic demands of the modern workforce.

Participants agreed that an assessment tool incorporating predefined steps and rubrics might effectively reduce bias, foster objectivity, and provide a transparent and structured framework for evaluating computational thinking-focused engineering projects. However, educator participants pointed out that presentation and collaboration skills require human evaluation. Participants emphasise that these skills involve nuances that machines may need to possess, mainly qualitative judgment. The sentiment expressed suggests a reliance on educators' understanding and experience to assess interpersonal and communication skills accurately. Moreover, student participants felt that educators possess a unique insight into student work and performance, highlighting the importance of their role in these assessments. Participants pointed out that lecturers will likely be more tolerant and understanding when assessing students. This leniency could be attributed to educators' personal rapport and insights into each student's circumstances, challenges, and efforts.

Considering the positive response and favourable comments received from educators and students during the initial inquiry, it is feasible to move forward with designing a Computational Thinking-Enhanced Assessment Tool (CTe-AT). The results strongly endorse the workability of this model, highlighting its capacity to improve the learning process and cultivate crucial problem-solving abilities. This optimistic reception forms a robust basis for the pragmatic creation and deployment of the CTe-AT, affirming its suitability as a valuable tool for evaluating and advancing computational thinking concepts within educational environments.

7. Limitations

A limitation of the preliminary study is its relatively modest sample size, comprising ten educators and ten students. The insights and findings drawn from this limited number of participants may only partially represent the broader population in computational thinking in engineering education. The small sample size limits the extent to which the study's results can be generalised, and the subtleties of perspectives within a more extensive and diverse group of educators and students need to be sufficiently captured. Consequently, it is advisable to exercise caution when extending the study's conclusions to a broader context. Further research involving a more extensive and diverse participant pool is suggested to enhance the study's external validity.

8. Conclusion

The necessity of designing assessment tools for engineering students centred on computational thinking is evident. This endeavour will better prepare graduates for the demands of the modern engineering workforce, foster critical thinking skills, and ensure effective alignment between theory and practice. Ultimately, this initiative will enhance the overall quality of engineering education and produce well-equipped graduates capable of addressing the challenges of the 21st century.

Declarations

Author Contributions. Nagaletchimee Annamalai- data collection, conceptualization, method design
Nasor - data analysis, introduction, literature review. All authors read and approved the final version.

Conflicts of Interest. The authors declare no conflict of interest

Funding. None

Ethical Approval. The study was conducted in accordance with the principles of the Declaration of Helsinki. Participants were informed about the objectives of the study and participated voluntarily, providing their informed consent. Measures were taken to ensure data anonymity, privacy, and

confidentiality.

Data Availability Statement. The data of this study is available from the corresponding author upon request.

Acknowledgements. None

References

- Anderson, L. W., & Krathwohl, D. R. (2001). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Addison Wesley Longman.
- Athanassiou, N., McNett, J. M., & Harvey, C. (2003). Critical thinking in the management classroom: Bloom's taxonomy as a learning tool. *Journal of Management Education*, 27(5), 533-555. <https://doi.org/10.1177/1052562903252515>
- Barak, M., & Usher, M. (2019). The innovation profile of nanotechnology team projects of face-to-face and online learners. *Computers & Education*, 137, 1-11. <https://doi.org/10.1016/j.compedu.2019.03.014>
- Braun, V., & Clarke, V. (2006), "Using thematic analysis in psychology", *Qualitative Research in Psychology*, 3(2), 77-101, doi: 10.1191/1478088706qp063oa.
- Dick, W., & Carey, L. M. (1996). *The systematic design of instruction*. New York, NY: HarperCollins.
- Falkner, K., & Sheard, J. (2019). Pedagogic approaches. In S. A. Fincher & A. V. Robins (Eds.), *The Cambridge handbook of computing education research* (pp. 445-467). Cambridge University Press. <https://doi.org/10.1017/9781108654555.016>
- Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38-43. <https://doi.org/10.3102/0013189X12463051>
- Howitt, D. (2012). Using qualitative methods to research offenders and forensic patients. In D. A. Crighton & G. J. Towl (Eds.), *Research in practice for forensic professionals* (pp. 132-158). Routledge.
- Kemp, S., Ng, M., Hollowood, T., & Hort, H. (2018). Introduction to descriptive analysis. In S. E. Kemp, J. Hort, & T. Hollowood (Eds.), *Descriptive analysis in sensory evaluation* (pp. 3-41). UK: John Wiley & Sons Ltd.
- Lee, J., Lim, C., & Kim, H. (2017). Development of an instructional design model for flipped learning in higher education. *Educational Technology Research and Development*, 65(2), 427-453. DOI 10.1007/s11423-016-9502-1
- Maxwell, J. (2005). *Qualitative research design: An interactive approach*. Thousand Oaks, CA: Sage
- Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: An expanded sourcebook* (2nd ed.). Sage.
- Patton, M. Q. (2002). *Qualitative research & evaluation methods*: Thousand Oaks: Sage Publications.
- Raaijmakers, J. G. W., & Shiffrin, R. M. (1981). Search of associative memory. *Psychological Review*, 88, 93-134.
- Rojas López, A., & García-Peñalvo, F. J. (2019). Personalised education for a programming course in higher education. In M. L. Sein-Echaluce, A. Fidalgo-Blanco, & F. J. García-Peñalvo (Eds.), *Innovative trends in flipped teaching and adaptive learning* (pp. 203-227). IGI Global. <https://doi.org/10.4018/978-1-5225-8142-0.ch009>

- Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in human behavior*, 72, 678-691. <https://doi.org/10.1016/j.chb.2016.08.047>
- Tsai, M.-J., Liang, J.-C., & Hsu, C.-Y. (2020). The computational thinking scale for computer literacy education. *Journal of Educational Computing Research*, 58(3), 687-709. <https://doi.org/10.1177/0735633120972356>
- Usher, M., & Barak, M. (2018). Peer assessment in a project-based engineering course: Comparing between on-campus and online learning environments. *Assessment & Evaluation in Higher Education*, 43(5), 745-759. <https://doi.org/10.1080/02602938.2017.1405238>
- Varela, C., Rebollar, C., García, O., Bravo, E., & Bilbao, J. (2019). Skills in computational thinking of engineering students of the first school year. *Heliyon*, 5(11), e02820. <https://doi.org/10.1016/j.heliyon.2019.e02820>
- Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Harvard University Press.

About the Contributor(s)

Nagaletchimee Annamalai, Department of Language and Culture, College of Humanities and Sciences, Ajman University, Ajman, UAE

Email: a.annamalai@ajman.ac.ae

ORCID: <https://orcid.org/0000-0002-8600-5624>

Mohamed Nasor, Department of Biomedical Engineering, College of Engineering and Information Technology, Ajman University, Ajman, UAE

Email: m.nasor@ajman.ac.ae

ORCID: <https://orcid.org/0000-0002-4833-4363>

Publisher's Note: The opinions, statements, and data presented in all publications are solely those of the individual author(s) and contributors and do not reflect the views of Universitepark, EDUPIJ, and/or the editor(s). Universitepark, the Journal, and/or the editor(s) accept no responsibility for any harm or damage to persons or property arising from the use of ideas, methods, instructions, or products mentioned in the content.
