Journal of Engineering Education Transformations,
Volume No. 37, January 2024 Special Issue, eISSN 2394-1707

Reflections and Decomposition in Problem Based Learning

Anusha Chikkamath[1], Prakash Hegade[2] and Ashok Shettar[3]
[1,2,3] KLE Technological University, Hubli, India
[1]01fe20bcs219@kletech.ac.in
[2]prakash.hegade@kletech.ac.in
[3]ashok@kletech.ac.in

***Abstract*—Problem Based Learning and Computational Thinking together have been applied in several studies, proven to be effective to bring out the best of problem solving. While both of them share several common features, computational thinking is a way of solving problems in problem based learning. One of the major components that direct a student towards critical thinking and self-directed learning are reflections. The components of computational thinking can be used to intervene reflections. This paper proposes to use decomposition for reflections. A multi method study was conducted on 37 students who were asked to write reflections for software architectural problems. The data was qualitatively analyzed using structural, in vivo and focused coding. Further, we used ttest to quantitatively analyze the data after receiving evaluation parameters from the qualitative analysis. The results are presented in this paper along with the themes generated and how reflections are effective when decomposition is explicitly stated. The work positively contributes towards design of effective PBL reflections.**

***Keywords*—Computational Thinking; Decomposition; Problem Based learning; Reflections.**

***JEET Category*—Research**

I. INTRODUCTION

MODULARITY is the strategic aspect majorly used in design problems, be it programming or system design. For an off-the-shelf component or designing a system with ease-of-use, modularity can help to achieve the separation of concerns. Most problem solving frameworks also emphasize on the same, to dividing the problem into manageable components, breaking them into smaller parts and then solving the problem. Every complex problem can be broken down into simpler workable components. In the language of Computational Thinking (CT), we call this process as decomposition (Wing, 2006). Decomposition is one of the four components, the other three being abstraction, pattern recognition and algorithms. CT in the recent times has been used as one of the learning frameworks. It is used along with the various available pedagogies and learning styles to enhance teaching and learning.

Education as we know is being entirely transformed to meet the needs of the present-day generation by preparing the students to become better critical thinkers. There is a need to build them with a professional skill that gives them an ability to solve complex real world scenarios. A shift towards Problem Based Learning (PBL) and experiential learning encourages the acquisition of practical skills application and critical thinking skills (Kolmos & De Graaff, 2007). As the process of learning becomes more student-centric, it emphasizes skill development and also the acquisition of knowledge (Barrows, 1998).

The PBL process involves students solving the real-world problems that are pertinent to their objectives and areas of interest, which encourages intrinsic motivation and self-determination. It necessitates that students keep track of their own learning progress, reflect on their methods and results, and change their course of action as necessary. This encourages self-regulation and metacognition. Reflecting on the learning process also acts as an added advantage to improve the knowledge gained and the ability to use this knowledge in the real world and to solve new problems. Reflection is a crucial part of the PBL approach. It is important because it encourages students to think about their experiences and find meaning in them. It makes them think at deeper levels and go beyond the problem being solved. It allows students to comprehend the problem they are working on. On the other hand, for a faculty, it is a means to understand if students have really understood the problem the way it was intended to be. Reflection increases the depth and richness of their learning. A variety of techniques, including journaling, group talks, peer evaluations, and self-evaluations, can help with this process (Procee, 2006).

Computational thinking is a problem-solving approach that entails decomposing complicated issues into smaller, more manageable chunks, spotting patterns and trends, coming up with algorithms and methodical approaches to tackle them, and then analyzing and improving those solutions. In a world filled with technology-driven complexity, the decomposition of the problem enhances efficiency as it allows for a methodical approach to problem-solving, which makes it simpler to identify trends, relationships, and potential solutions. As PBL can encompass any of the methods to solve a problem, CT can be one of them. This work proposes to understand the role of reflections designed with respect to decomposition in a PBL setting.

The paper is further divided into the following sections:

section II presents the literature survey, section III presents the research design, Section IV presents the methodology, section V presents the results and data analysis, section VI presents the discussion followed by conclusion in section VII.

II. Literature Survey

This section reviews the literature in the fronts of PBL, CT, decomposition and reflections. PBL is one of the most innovative methods designed to improve the learning process of a student by developing a need to solve real-life problems along with the learning process. It aims to develop the students' problem solving skills and empower them to become self-directed learners and effective critical thinkers. Initially introduced in the medical field, PBL has now found its way into the engineering field to transform the way of traditional teaching methods (Chen et al., 2021). PBL has also been explored with one day many problems approach (Hegade, 2019).

The focus on active, student-centered learning distinguishes PBL from other educational approaches. There is a reciprocal relationship between knowledge and the problem since the information and abilities to be learned are organized around problems rather than as a hierarchical list of topics (Hung et al., 2008). PBL proves to be effective as students work collaboratively in groups thereby improving their communication skills and the ability to work within a team. Self-directed learning is achieved as students take ownership of their education by conducting research, identifying learning objectives and proposing solutions (Loyens et al., 2008). An interdisciplinary approach along with continuous assessment and feedback supports holistic learning and prepares students to solve the real-world problems. Industry collaborated reflections and scaffoldings have been explored in the regards (Patil et al., 2023).

A vital connection between gaining new information and applying it in meaningful ways does exist and reflection can bring it out in the learning process (Turns et al., 2014). Reflection enables students to take a step back and examine their entire learning process as well as appreciate the value of their information as a whole rather than simply in isolated knowledge chunks (Chang, 2019). It is an effective tool for self-evaluation and skill development since it allows one to discover their areas of strength and need for progress. Additionally, by urging students to think about practical applications, it links theory to practice and improves information retention over time (Mortari, 2015).

CT is a skill that enables people to solve problems in a systematic and efficient way, using human thinking and computer as a tool (Wing 2006). Wing defines CT as the ability to abstract, automate, analyze, and generalize problems and solutions using computational models and tools. Pattern recognition, decomposition, abstraction, and algorithmic design are key components of computational thinking, which help people come up with efficient and effective solutions to problems. It can be taught and learned at different levels of education, from kindergarten to university and beyond which can help students become more confident, competent and independent learners who can adapt to the changing demands of the 21st century (Cansu & Cansu, 2019). Decomposition is one of the core skills of CT that involves breaking down a complex problem into smaller and more manageable parts (Rich et al., 2019). This method makes it simpler to solve, manage, or automate problems by enabling people to study and comprehend the numerous components of a problem. Decomposition can be effective in problem solving (Egidi, 2006) and it is one of the well-known problem solving strategies (Gick, 1986). Nevertheless, CT has been promoted through projects with several research works (Shin et al., 2021). PBL has been discussed as one of the facilitators of the computational thinking (Jonasen & Gram-Hansen, 2019). There is systematic review studies that present the works of CT integrated with PBL (Salam, 2022). Decomposition has been explored with game based learning (Hegade et al., 2023).

Socrates affirms it's not a human life if there is no reflective thinking and this has to be done in the educative process. According to Dewey, thinking occurs when things are uncertain or doubtful or problematic (Dewey, 1916). And reflective practice can help one be self-aware and raise a thoughtful eye (Dahlberg et al., 2002). Literature presents an opportunity to combine the reflections with decomposition problems and inspect their effects on student learning and cognitive skills.

III. Research Design

This section presents the employed research design for the proposed work.

A. Interpretive Framework

The interpretive framework considered for the study is pragmatism (Creswell & Poth, 2013). Under this, we define our ontology, epistemology, axiology and methodology. The ontological beliefs are that reflections are beneficial and evaluating their effectiveness can provide means on how to design them. Inductive approach is used for knowledge construction. The work has its defined limitations as the data collected for analysis is from one class from a single considered demography. The work presents multiple realities by combining the perspectives of both the participant and the researcher. The work uses multi method approach for data analysis. Both qualitative and quantitative methods are used and appropriate tools are selected as per need.

B. Research Question

Considering the gaps and fields of study interest, which are PBL and CT, we construct the research question as

RQ: How are reflections different when computational thinking's decomposition is integrated with problem based learning case studies?

We explore this by designing a case study which explicitly asks to decompose and reflect and other which only asks to reflect. Both cases naturally will have decomposition in them, one being explicit and the other not.

C. Model

Based on the gaps in literature survey and the elements of study under consideration, a model was designed to guide the case study design. This model is presented in Fig. 1.

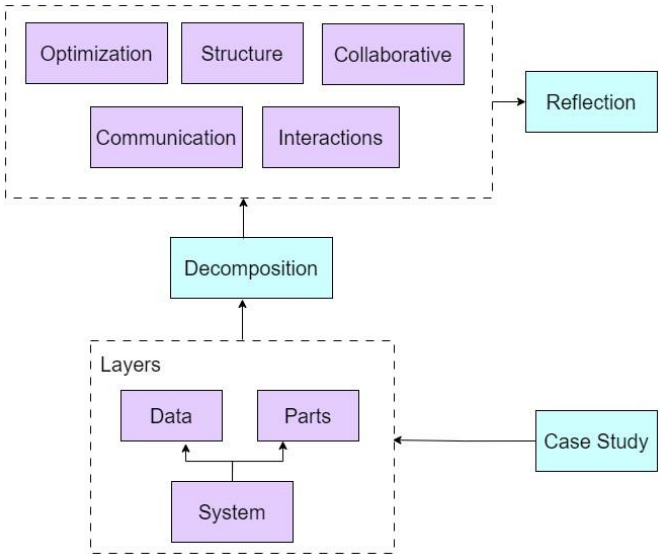

Fig. 1: Model for case study design

The case studies are designed to incorporate a system that has data and parts. These are spread across layers and more than one. This layer is what constitutes decomposition. Decomposition constitutes of structure, interactions, communication, optimization and collaborations. It involves breaking down complex problems into manageable parts, emphasizing structured analysis, effective communication, optimization of individual components, and fostering collaborative efforts for a holistic understanding and solution development. Reflections are formulated based on these criterions.

D. Sampling

The sampling method used was purposive sampling (Etikan et al., 2016). The entire population was chosen for sampling to avoid bias in the results (Sharma, 2017). Data was collected from 37 students undergoing the course Model Thinking with Course Code Knit-101 from the School of Computer Science and Engineering department. A consent form was shared with all the students stating that the data will be used for research purpose. With their signed consent and permission, as per university guidelines, data was collected for the research work. The data collection had two parts. One was the survey form and another was submitting case study solution sheets in document format.

IV. Method

This section describes the objective and the case study design for the data collection.

A. Objectives

The objectives of the work are as listed below in Table 1 formulated on the basis of the research question design.

TABLE I
OBJECTIVES

ID.	Objective
O_1	To understand the decomposition present in the system
O_2	To reflect on the decomposition judgmentally relating to the real-world applications
O_3	To identify the conceptual themes in the reflections

The objectives were used as a guide to formulate the case study and questions for reflection.

B. Architecture Case Study Design

As decomposition is supposed to be the emphasis for the reflections, architectural pattern problems were selected for the case study design. The questions are listed below.

Question 1: What do you mean by a full stack web development? Write your reflection. Question 2: India Stack Architecture can be decomposed into 4 layers. Study the India Stack Architecture and understand its significance. Layered design is basically decomposition. Write your reflections on the four layers. As a responsible engineer, how can you contribute to this architecture?

Though both problems require decomposition and writing reflections, the second one is more explicit than the first one.

V. Results and Data Analysis

This section presents the results and the data analysis. The solutions presented by the students were graded for the quantitative study. The results are presented in the following sub-sections. Two rounds of coding were carried out on the case study data for qualitative analysis. In the first round structural and in vivo coding was used. Structural Coding usually helps in identifying large segments of text on broad topics and these segments can then form the basis for an in-depth analysis within or across topics (Saldana, 2021). For in vivo coding, we use the exact phrases and words collected from the case study answers and perspectives (Manning, 2017). Focused coding was employed for the second round where we categorized the related themes and merged them (Benaquisto & Given, 2008). The paper presents the summary and analysis of the coding methods. The data was quantitatively analyzed using a ttest.

A. First Round Coding

Both structural and in vivo coding were used in the first round. As the name suggests, structural coding allows us to code each of the answers based on distinct sections within the document, i.e. based on how the answers are structured. Considering an example of a paper based on any given topic, structural coding can be performed by providing centric labels like “methodology,” “literature review” and “hypothesis” initially and then coding each section at a time which will make it easier for us to navigate and refer to sections quickly, saving us from having to go through data sections again. We can consider another example of a survey to be conducted based on eating habits and cultivating healthy eating habits. For this, the segments identified would be “previous eating habits”, “motivation to begin”, “helpful techniques” and “impact of program”.

In our method, the segments identified were "architecture", "engineer role", "security" and "real-life examples". The answers of the students were coded based on these segments, emphasizing one segment at a time and acquiring codes. For example, the sentence from the answer was "Layered architecture signifies abstraction by keeping detailed information hidden". We coded this into "abstraction through layers" using structural coding emphasizing the 'architecture' segment. Similarly, we coded "traditional forms of identification to a more inclusive and efficient" sentence into "Modernizing Identification" while coding the 'security' related segment of the answer. An example for coding the 'engineer role' segment would be "Engineer can design the processes that save time and effort" coded into "Efficient Process Design". Similarly, we coded the sentence "The stack model with Aadhar facility, e-KYC, should be available throughout the year providing no constraint in time to access" into "Continuous Availability" while referring to the 'real-life example' segment of the answer.

In vivo coding, we remain as close to the participant's words as possible while coding. Considering a general example sentence "The fact that I've made the analysis so extensive has only increased my anxiety about what I need to do", this can be coded into "Increased anxiety" using vivo coding by fetching words from the original sentence itself. In vivo coding can prove to be efficient for us as we don't deviate from the topic as answered by the student and remain close to the student's answer and not induce any bias.

For example, the sentence from the answer was "Building flexible models that are adaptable to future needs". We coded this into "Flexible, Adaptable Models" using vivo coding. Similarly, "Having a thorough knowledge of stack being used helps in improved communication" was coded into "Improved Communication". Another example can be the sentence "Indian stack helps to gain attention by properly decomposing tasks and solve issues by providing better services" which we coded into "Task decomposition and QOS".

Answers were analyzed to identify the themes. A sample of few themes assigned is presented in Table 2 below. Several such themes were classified from the case studies. 32 such reflections were identified from the case studies.

TABLE II
SAMPLE THEMES AND CODES

Student Reflection	Theme	Code
Building flexible models that are adaptable to future needs	Model development	Flexible, Adaptable Models
Collaboration is key for creating a cohesive and effective system	Model development	Collaborative synergy for systems
Complex applications are built from simple pages	Management	Progressive complexity
Layered architecture signifies abstraction by keeping detailed information hidden	Data handling	Abstraction through layers

The summary of themes identified is presented in Table 3 below from the set of 32 identified reflections. All others saturated into one of the 32 identified reflections. A few of them have been merged as the intention behind the task was similar.

TABLE III
THEMES CATEGORIES

Themes	Codes
Authorization	Modernizing identification Effective service providence Task decomposition and QOS Improved authentication in-component parts
Communication	Revolutionizing system-interaction Continuous availability Communication channels Improved Communication
Data handling	Data optimization and-structuring Data security Data privacy and control Abstraction through layers
Management	Multi-Tech proficiency Progressive complexity Comprehensive management Dual role management
Model development	Flexible, Adaptable Models Efficient process design Collaborative synergy for -System efficiency Layered problem solving Isolation through-decomposition Improved flexibility

B. Second Round Coding

Coding is actually a cyclical act. The second round of coding is usually performed to focus on the salient features for generating concepts, grasping meaning, and/or building theory (Saldana, 2021). In second round coding, from the themes and codes, the conceptual mappings were identified. These were mapped based on computer science and engineering conceptual understanding. Focused coding allows us to refine our initial coding pass into a final set of codes and categories. Its advantage lies in its efficiency, allowing us to concentrate on key issues, facilitating a deeper understanding of the central concepts within the data. For example, we mapped the codes "Flexible, Adaptable models" and "Collaborative synergy for systems" to the theme 'Model development'. Similarly, "Data optimization and- structuring", "Data security" and "Data privacy and control" were together

mapped to 'Data handling' as they all talk about the data. The codes "Modernizing identification", "Effective service providence", "Task decomposition and QOS" and "Improved authentication in-component parts" were mapped to 'Authorization' as these are related to authentication and authorization.

With focused coding, relationships across the different themes were identified to come up with evaluation parameters. Table 3 represents the themes identified in second round coding through the codes generated in first round. We saw that the students mainly focused on the properties of seamless authorization, efficient communication, effective data handling, streamlined management and model development while answering the questions.

Further, this helped us to come up with evaluation parameters. The theme model development helped us to understand if the student has actually related to the past concept and how the model was designed. The themes management and data handling summed up to how well the students understood the problem. The themes authorization and communication were included to come up with the parameter of listing applications as students included several real-time applications of providing authorization and communication between layers. Parameters were decided based on decision questions, how well the architecture was understood, if they related to the known concepts and if they had listed applications. These parameters seemed apt to evaluate the answers of the students. The themes appeared to be interconnected and cross the different themes.

C. Case Study Scores

Table 4 presents the evaluation of 37 answers based on the identified parameters of second round coding process.

TABLE IV
SCORES ON IDENTIFIED PARAMETERS

Section	Full stack	Indian stack
Decision question	0.964	1
Understanding of the problem	3.122	3.36
Do they relate to the past concept	2.54	2.78
Listed applications	2. 83	2.89

Table 5 below presents the overall scores of two case studies designed.

TABLE V
OVERALL SCORE

Section	Full stack	Indian stack
Average	14.81	15.81
Variance	3.13	2.93
Standard Deviation	1.74	1.71

The two studies were validated using the ttest using the spreadsheet application. The formula was used to produce the results. The hypotheses were written as

Null: two group means are equal.

Alternative: two group means are not equal.

The results obtained can be seen in Table 6 below.

TABLE VI
TTEST RESULTS

T-Statistic	Scores
P(T<=t) two-tail	0.021074968
T Critical two-tail	1.994437086

As the p value is less than the value of significance (two tailed test), we hence reject the null hypothesis and accept the alternative hypothesis. We can thus conclude that the two group means are not equal.

D. Student Feedback

The feedback was collected from 37 students on two questions. One was on overall feedback for the course and the second one was the learning that happened with the decomposition case study. The two feedbacks are presented in Fig. 2 and Fig. 3 respectively below.

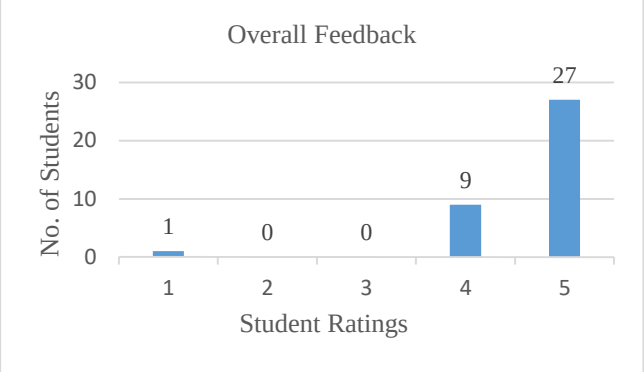

Fig. 2: Overall course feedback

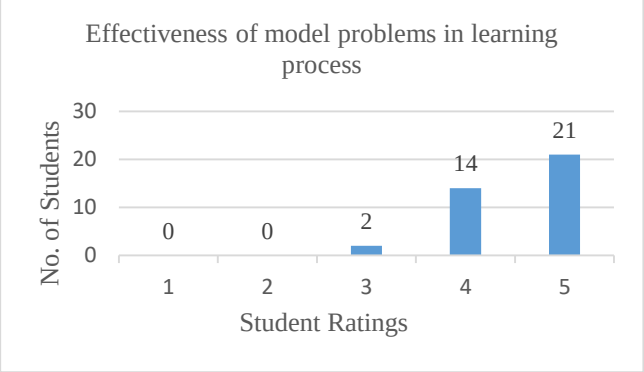

Fig. 3: Learning effectiveness of case study

We can note from the feedback given by 37 students that the learning process has been satisfactory.

VI. Discussion

This case study helps us to arrive at stimulating conversation points. Though the mean values are almost the same, and we have concluded that the two group means are not equal. To explore this further, the themes generated were revisited further to analyze each of the case studies. Table 7 below presents the number of themes that originated from each case study. The table presents the themes after combining the relevant ones.

TABLE VII
THEMES FROM EACH CASE STUDY

Type	Number of Themes
Full Stack	8
India Stack	14
Common to Both	5

The numbers clearly show that when decomposition is explicitly stated, the number of themes generated is more (14 + 5). Some of the major themes like Model development appeared from the second case study. When decomposition was reflected, there were more themes that interconnected with the systems students had designed. Issues of communication and data handling were addressed. What this study indicates is that, with similar mean scores and similar assessment statistics, the learning and take away when decomposition is explicitly asked is higher than when not asked. When a key indicator is stated for reflections, students try to comprehend a system not only in parts but also on how they work together to be a holistic system. Reflections provide a means to think in terms of an interactive system. Both individual components of a system and the system as a whole can be properly analyzed by the students through decomposition which will help them to better understand the problem given to them and apply it in real-life scenarios.

Student feedback has been positive indicating that the process has been effective in their learning. 35 of 37 students were satisfied with the case study learning as they have rated satisfied or highly satisfied with respect to learning process.

VII. Conclusion

Decomposition from CT can be combined with PBL to design effective reflections. The objective of this work was three-fold. The first one was to understand the decomposition aspect of the system. By selecting and designing architecture case studies, this was achieved. However this is also applicable to any kind of case study formulation. Decomposition was further related with respect to real world problems. Students critically provided reflections on the decomposed system. To identify the conceptual themes in the reflections, we coded the data and identified that decomposition can provide meaningful themes and help to ease the understanding of any complex problem. It can bring in several conceptual themes through reflections which was evident in our study that we conducted. This promises to be an effective way to design PBL reflections.

References

Barrows, H. S. (1998). The essentials of problem-based learning. *Journal of Dental Education*, *62*(9), 630-33.

Benaquisto, L., & Given, L. (2008). The SAGE encyclopedia of qualitative research methods. *Given L, ed*, *413*.

Cansu, F. K., & Cansu, S. K. (2019). An overview of computational thinking. *International Journal of Computer Science Education in Schools*, *3*(1), 17-30.

Chen, J., Kolmos, A., & Du, X. (2021). Forms of implementation and challenges of PBL in engineering education: a review of literature. *European Journal of Engineering Education*, *46*(1), 90-115.

Chang, B. (2019). Reflection in learning. *Online Learning*, *23*(1), 95-110.

Creswell, J. W., & Poth, C. N. (2013). Philosophical assumptions and interpretive frameworks. *Qualitative inquiry and research design: choosing among five approaches. Los Angeles: Sage Publications*, 15-41.

Dahlberg, K., Drew, N., & Nystrom, M. (2002). Reflective lifeworld.

Dewey, J. (1916). Democracy and education. New York, NY: The Macmillan Company research. Lund, Sweden: Studentlitteratur.

Egidi, M. (2006). Decomposition patterns in problem solving. *Contributions to Economic Analysis*, *280*, 15-46.

Etikan, I., Musa, S. A., & Alkassim, R. S. (2016). Comparison of convenience sampling and purposive sampling. *American journal of theoretical and applied statistics*, *5*(1), 1-4.

Gick, M. L. (1986). Problem-solving strategies. *Educational psychologist*, *21*(1-2), 99-120.

Hung, W., Jonassen, D. H., & Liu, R. (2008). Problem-based learning. Handbook of research on educational communications and technology, 3(1), 485-506.

Hegade, P. (2019). One-Day Many-Problems: A Problem Based Learning Approach. *Journal of Engineering Education Transformations*, *33*(1), 1.

Hegade, P., Patil, O., & Shettar, A. (2023, February). Effectiveness of Game Based Learning in Problem Based Learning. In *2023 2nd Edition of IEEE Delhi Section Flagship Conference (DELCON)* (pp. 1-6). IEEE.

Jonasen, T. S., & Gram-Hansen, S. B. (2019). Problem based learning: A facilitator of computational thinking. In *Proceedings of the 18th European Conference on e-Learning: ECEL* (pp. 260-7).

Kolmos, A., & De Graaff, E. (2007). Process of Changing to PBL. In *Management of change* (pp. 31-43). Brill.

Loyens, S. M., Magda, J., & Rikers, R. M. (2008). Self-directed learning in problem-based learning and its relationships with self-regulated learning. *Educational psychology review*, *20*, 411-427.

Manning, J. (2017). In vivo coding. *The international encyclopedia of communication research methods*, *24*, 1-2.

Mortari, L. (2015). Reflectivity in research practice: An overview of different perspectives. *International journal of qualitative methods*, *14*(5), 1609406915618045.

Patil, O., Hegade, P., & Shettar, A. (2023). Industry Collaborated Reflections and Scaffolding in Problem Based Learning. *Journal of Engineering Education Transformations*, *36*(Special Issue 2).

Procee, H. (2006). Reflection in education: A Kantian epistemology. *Educational theory*, *56*(3), 237-253.

Rich, P. J., Egan, G., & Ellsworth, J. (2019, July). A framework for decomposition in computational thinking. In *Proceedings of the 2019 ACM conference on innovation and technology in computer science education* (pp. 416-421).

Salam, S. (2022). A systemic review of Problem-Based Learning (PBL) and Computational Thinking (CT) in teaching and learning. *International Journal of Humanities and Innovation (IJHI)*, *5*(2), 46-52.

Saldaña, J. (2021). *The coding manual for qualitative researchers*. sage.

Shin, N., Bowers, J., Krajcik, J., & Damelin, D. (2021). Promoting computational thinking through project-based learning. *Disciplinary and Interdisciplinary Science Education Research*, *3*(1), 1-15.

Sharma, G. (2017). Pros and cons of different sampling techniques. *International journal of applied research*, *3*(7), 749-752.

Turns, J. A., Sattler, B., Yasuhara, K., Borgford-Parnell, J. L., & Atman, C. J. (2014, June). Integrating reflection into engineering education. In *2014 ASEE Annual Conference & Exposition* (pp. 24-776).

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.