

https://dspace.org/

Semiotic analysis of UML: class diagrams in pedagogy: a case of first-year BCIS students

Pakiso Khomokhoana; R. C Fouche; T. S Nkalai

https://hdl.handle.net/10539/46643
SCOPUS

Downloaded from a DSpace repository. https://dspace.org/

Discover Education

RESEARCH Open Access

Semiotic analysis of UML class diagrams in pedagogy: a case of first-year BCIS students

Pakiso J. Khomokhoana[1*], Rouxan C. Fouché[2] and Tlholohelo S. Nkalai[2]

*Correspondence:
Pakiso J. Khomokhoana
pakiso.khomokhoana@wits.ac.za
[1]School of Business Sciences, University of the Witwatersrand, Johannesburg, South Africa
[2]Department of Computer Science and Informatics, University of the Free State, Bloemfontein, South Africa

Abstract

Unified Modelling Language (UML) class diagrams are standard tools in software engineering education, typically analysed for syntactical correctness rather than their communicative dimensions. This study applies semiotic theory to investigate how first-year Bachelor of Computer Information Systems students engage with UML class diagrams as communicative artifacts. Using a narrative research approach based on the integrated-methods research framework, data was collected through individual semi-structured interviews incorporating think-aloud protocols with 11 participants. Participants completed tasks of increasing complexity: analysing simple diagrams, describing relationships between classes, and creating their own class diagrams. Data analysis followed a systematic coding framework aligned with the six semiotic principles: identifying diagram elements, deciphering signs and symbols, understanding the contextual significance, interpretation and meaning, contextualisation within broader discourse, and critical reflection. Findings reveal that students employ diverse strategies when approaching UML class diagrams, with consistent analytical approaches yielding better comprehension. Visual elements significantly influence interpretation, while terminological confusion and cognitive load present key challenges. This research contributes to UML pedagogy by highlighting the need for educational approaches that address both technical notation and interpretive strategies, balancing concrete examples with abstract structural thinking.

Keywords Information systems education, Unified modelling language (UML) class diagrams, Semiotic theory, Principles of semiotic analysis, Communicative elements of UML class diagrams, UML class diagram pedagogy

1 Introduction

Diagrams serve as powerful communication tools that often bridge gaps that written descriptions cannot fully address [24, 30, 43]. In disciplines such as Information Systems (IS) and software engineering, Unified Modelling Language (UML) diagrams have become standard visual representations for complex systems [12, 19, 42]. These diagrams transcend mere structural and syntactical roles; they function as communicative artifacts through which designers express their understanding of systems, share design concepts, and convey intentions to peers and stakeholders [17].

© The Author(s) 2025. **Open Access** This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

When students create UML class diagrams, they engage in a communicative act that extends beyond following syntactic rules. These diagrams enable students to express their conceptual understanding, problem-solving approaches, and interpretations of system requirements [1]. Despite their important role as communicative tools, traditional pedagogical approaches often emphasise syntactical correctness while overlooking the deeper communicative dimensions of UML diagrams [7, 37].

UML class diagrams are particularly valuable for transforming complex textual descriptions into precise visual models [19, 38, 47, 49]. While previous research has established that these diagrams effectively convey ideas challenging to express in text [24, 30, 43], a significant gap exists in understanding how students interpret and utilise these diagrams as communicative tools. Specifically, limited research exists on how the semiotic elements of UML class diagrams—their signs, symbols, and cultural significance—contribute to learning outcomes and comprehension in IS and software engineering education [6, 44].

Semiotics, the study of signs and meaning-making, offers a structured framework for analysing how UML diagrams function as communicative artifacts beyond their technical specifications [3, 11, 36]. This theoretical approach enables researchers to examine the complex ways in which visual elements convey meaning within specific educational and professional contexts.

This study aims to address the following research questions:

- What semiotic elements should be considered when designing pedagogical strategies that involve UML class diagrams?
- In working with UML class diagram-based activities, what are participants communicating through the lens of semiotic theory?
- How can semiotic analysis of these elements enhance broader pedagogical approaches for teaching UML class diagrams in IS courses?

This study extends previous UML pedagogy research by adopting a semiotic theoretical lens. While Khomokhoana and Nkalai [23] examined cognitive processes students use when learning UML concepts, this study investigates how UML class diagrams function as communicative artifacts through semiotic analysis. This shift from cognitive processes to communicative semiotics represents a fundamentally different theoretical approach to understanding UML pedagogy. This study focuses on the pedagogical dimension of UML class diagram interpretation in educational contexts. While professional UML practice emphasises technical accuracy, pedagogical use prioritises conceptual understanding and communicative competence development. Educational settings reveal emerging semiotic strategies before they become automated through professional practice.

This research focuses on first-year Bachelor of Computer Information Systems (BCIS) students, examining how novices interpret and apply modelling conventions before fully adopting professional practices. Their strategies and challenges provide valuable insights into the learning process, often overlooked in more experienced practitioners.

The study captures students' cognitive and communicative strategies while working on UML class diagram tasks using a narrative research approach with think-aloud protocols, observations, and artefact analysis. The findings enhance UML class diagram pedagogy and semiotic theory, offering insights for effective teaching and a deeper

understanding of visual communication in technical contexts. This research bridges the gap between technical correctness and communicative effectiveness in UML education.

2 Theoretical background and literature review

2.1 UML class diagrams as communicative tools

UML class diagrams operate within broader software architecture frameworks, including architectural views models 1+5 by Górski [20, 25] and 4+1 by Kruchten [24] which position structural modeling as fundamental to system design [20–21, 25]. This architectural context emphasises the importance of UML class diagrams as communicative tools that must effectively convey structural relationships to diverse stakeholders in software development processes.

UML class diagrams are key tools in software engineering and systems design, offering visual representations that simplify complex processes by converting detailed text into understandable models [19, 47, 49]. Traditionally, their purpose has been viewed through a technical lens, focusing on accurately representing system structures and behaviours for development.

Research in UML class diagrams education has mainly concentrated on assessing students' technical competence in creating syntactically correct diagrams [13, 16]. Scholars like Booch et al. [5] and Larman [26] stress teaching syntactical rules and accurate system representation. While essential, this focus only addresses one aspect of UML class diagrams' skills.

More recent studies recognise UML diagrams as tools for communication, not just technical artifacts. France and Rumpe [17] note that UML class diagrams enable designers to convey their understanding and share intentions with peers and stakeholders. Similarly, Abrejo et al. [1] suggest that UML class diagrams reflect students' conceptual understanding and problem-solving approaches. Despite this, the communicative aspect of UML class diagrams is still underexplored, especially in educational settings where students are refining their modelling skills [6]. This gap is particularly pronounced in educational contexts, where students simultaneously develop technical skills and communicative competence. Unlike professional practitioners with automated UML conventions, students provide unique insights into the fundamental semiotic processes underlying diagram interpretation.

UML class diagrams reduce software engineering complexity by turning detailed text into precise visual models [19, 47, 49]. Literature [23, 30, 43] suggests that diagrams help convey concepts that are challenging to express in text. Despite their communicative role, traditional analyses often focus mainly on correctness and completeness, overlooking their broader communicative function [7, 37].

2.2 Semiotic theory as an analytical framework

This study uses semiotic theory [3, 11, 36] as an analytical framework to explore the communicative dimensions of UML diagrams. Semiotics is the study of signs, symbols, and their interpretation [7, 31, 45], offering a structured approach to understanding how diagrams convey meaning within specific contexts [50]. Ferdinand de Saussure and Charles Sanders Peirce founded modern semiotic theory. De Saussure [11] described signs as having two parts: the signifier (the form) and the signified (the concept), with their relationship being arbitrary and shaped by convention and cultural context. Peirce

[36–37] expanded this idea with a triadic model involving the sign, object, and interpretant, focusing on the interpretive process in meaning-making. Roland Barthes [3] further developed semiotics by examining how signs function within broader cultural systems. He elaborated on concepts like denotation (literal meaning) and connotation (cultural associations). He argued that meaning arises from the interaction between the text and the reader, not from the author's intent. Applied to UML class diagrams, semiotic theory provides a framework for understanding how these diagrams act as systems of signs to communicate meaning. Every element in a UML class diagram—from shapes and lines to the arrangement of components—can be analysed as a sign contributing to the overall meaning of the diagram.

Semiotic theory provides a framework for understanding UML class diagrams as systems of signs that communicate meaning. Every element, from shapes to layout, contributes to the diagram's overall meaning. Siau and Tian [45] demonstrated that iconic notations, which visually resemble their meaning, are more accurately interpreted and evoke fewer connotations than symbolic notations. Their research highlights semiotics as a valuable tool for evaluating UML diagrams' effectiveness as communication tools.

2.3 Principles of semiotic analysis

Based on the works of semiotics scholars [3, 7, 11, 36], this study utilises the six core principles of semiotic analysis as a framework to investigate the communicative aspects of UML class diagrams.

2.3.1 Principle 1—identifying diagram elements

This principle involves recognising the basic components and understanding their broader system context [3, 7, 11, 36]. These elements include shapes, lines, colours, symbols, labels, and other visual components. In UML class diagrams, they represent classes, attributes, and relationships [12, 38, 41].

In UML class diagrams, students typically identify classes (e.g., Student or Computer), determine attributes (e.g., Student Name or Computer Memory), and define relationships (e.g., one-to-many) [5, 26]. However, students sometimes confuse elements across different diagram types, such as using lifelines or activation bars from sequence diagrams in class diagrams [16]. This highlights the importance of understanding the diagram's purpose. Studies show that students often focus on visible cues like shapes and symbols, but overlook their semantic meaning, leading to design errors [13]. For instance, incorrectly associating a class like Computer with a student using a composition relationship instead of an association indicates a lack of contextual understanding. Understanding the data source and intended purpose of the diagram is crucial, as these contextual factors influence how students identify and prioritise different elements within the visual representations.

2.3.2 Principle 2—deciphering signs and symbols

UML class diagrams use symbols and notations with specific meanings that contribute to the overall message [2–3, 11, 37]. Shapes and lines represent structure or relationships, while colours highlight key elements or evoke emotions [34].

Students rely on standard notations to represent system components. For instance, rectangles represent classes, with sections for names, attributes, and methods [5, 49].

Lines and arrows indicate relationships, with arrowheads and line types specifying the connection [26, 42]. However, students sometimes misuse these symbols, leading to unclear diagrams [13]. For example, a dashed arrow meant for dependencies might be used instead of a solid arrow for associations, altering the diagram's meaning [16, 43]. These challenges with symbol interpretation align with empirical research by Siau and Tian [45], who found that UML notations that are more symbolic (rather than iconic) are interpreted less accurately by users, highlighting the importance of visual resemblance in notation design. Though uncommon in traditional UML, students sometimes use colour to emphasise elements, such as distinguishing user-related from system-related classes. While this can enhance readability [29], it may lead to inconsistency if the colour used is not standardised or explained in the diagram's legend.

2.3.3 Principle 3—understanding cultural and contextual significance

The meaning of signs and symbols in UML class diagrams can vary based on cultural and contextual factors. Students' backgrounds and experiences influence how they interpret diagrammatic elements [3, 11, 14, 36]. For instance, students from different educational or cultural backgrounds may interpret UML notations differently, depending on prior exposure.

Dobing and Parsons [13] found that students often bring preconceived notions to diagrams, leading to varied interpretations of symbols like arrows or dashed lines. For example, students unfamiliar with UML may misinterpret a dashed arrow (used for dependencies) as indicating a weaker relationship than a solid arrow, hence misunderstanding its technical meaning. The use of colours, shapes, and visual arrangements can reflect personal or cultural preferences, affecting diagram clarity. Moody [29] noted that students often use colour to emphasise components, but its meaning may differ among viewers. One student might use red to signify errors, while another uses it to mark high-priority classes, leading to confusion in collaborative interpretations. Students' experiences with other diagramming tools also shape their approach. For example, those familiar with flowcharts or entity-relationship diagrams may inadvertently incorporate elements like decision nodes or keys into UML class diagrams, creating hybrid representations [16].

2.3.4 Principle 4—interpretation and meaning

The interpretation and meaning principle explores how blocks and their arrangements are understood, potentially revealing misunderstandings or alternative interpretations that differ from the designer's intent. This aligns with *Principle 3 (understanding cultural and contextual significance)*, emphasising that meaning is influenced by the viewer's perspective. Barthes' "*Death of the Author*" theory asserts that meaning emerges from the interaction between the viewer and the work, not solely from the creator's intent [3].

Students often design diagrams with specific meanings in mind, but these can be misinterpreted due to symbol ambiguities or layout issues. Dobing and Parsons [13] found that students sometimes misuse generalisation arrows (open triangle heads) to represent associations, misinterpreting class hierarchies as simple relationships. Alternative interpretations also arise from inconsistencies in diagram organisation. Fowler [16] observed that students often place elements arbitrarily, which can confuse viewers about system modularity or dependencies. The omission of critical information or overloading

symbols with multiple meanings exacerbates these issues. Moody [29] emphasised the need to reduce "*cognitive load*", but students often fail to simplify complex relationships. For example, representing multiple relationships with various lines and annotations can obscure the primary connection. Misinterpretations are also influenced by the viewer's familiarity with UML class diagrams' conventions. Novices may rely on visual cues like shapes and arrows rather than textual labels, leading to incorrect assumptions about diagram elements [5]. For example, a dashed line in a sequence diagram may be misinterpreted as a weaker connection rather than a message.

2.3.5 Principle 5—contextualising within larger discourse

Diagrams are part of a broader discourse in IS education, and this principle explores how UML class diagrams connect to other texts, images, or cultural phenomena. Analysing their role within larger narratives helps uncover deeper meanings and their significance in cultural and social contexts [3, 11, 36].

Students often draw on prior knowledge or external sources when creating UML class diagrams, but this intertextuality can lead to unintended interpretations. Dobing and Parsons [13] found that students frequently rely on textbook templates, replicating errors or outdated conventions. While influenced by educational discourse, these diagrams may not align with industry standards, causing a disconnect between academic and professional contexts. Cultural and social backgrounds also shape the symbols and layouts used in UML class diagrams. Moody [29] observed that cultural conventions, like reading direction, affect diagram comprehension. For instance, students from right-to-left reading cultures may arrange diagrams differently, altering how relationships and flows are perceived. UML diagrams also interact with other forms of communication, such as textual specifications or verbal explanations. Larman [26] noted that UML class diagrams are often paired with accompanying documentation, which students may neglect or fail to align with, reducing clarity and interpretability within the broader discourse. Additionally, UML class diagrams play a role in educational narratives, shaping how students communicate their understanding to instructors, peers, or evaluators [16].

2.3.6 Principle 6—critical reflection

This principle involves analysing diagrams and findings, considering their implications, and reflecting on any biases or assumptions that may influence their interpretation. This approach ensures a nuanced and thoughtful analysis [3, 11, 36]. In UML class diagrams, students often struggle to assess completeness and accuracy. Dobing and Parsons [13] found that students overlook inconsistencies, such as mismatched relationships or undefined attributes, focusing on surface-level correctness instead of system coherence. Critical reflection helps students identify these oversights and align their diagrams with the intended design. Biases influence diagram construction and interpretation. Fowler [16] observed that students often overemphasise class hierarchies due to preconceptions about object-oriented design. These biases may lead to overly complex diagrams or omit key components like behavioural aspects.

Reflection promotes a holistic approach and prevents skewed interpretations. Critical reflection is vital for considering the broader implications of design choices. Moody [29] noted that visual notations inherently influence interpretation, such as when students unconsciously prioritise elements based on size or placement. Reflecting on these

biases ensures diagrams accurately communicate intended priorities and relationships. Additionally, critical reflection addresses implicit assumptions. Larman [26] pointed out that students often assume viewers share their technical knowledge, leading to diagrams lacking adequate annotations. Reflecting on design accessibility encourages students to create more inclusive and comprehensible diagrams, improving communication within diverse teams. In UML class diagrams, critical reflection involves assessing diagram completeness and semantic accuracy. This verification process reveals students' metacognitive awareness and recognition of diagrams as communicative artifacts requiring systematic evaluation.

These six principles provide a comprehensive framework for analysing UML diagrams beyond technical correctness, focusing on their role as communicative artifacts in educational contexts.

2.4 Semiotic theory to UML education

Applying semiotic theory [3, 11, 36] to UML class diagrams' education offers a fresh perspective on how students learn and apply modelling concepts. Traditional UML class diagrams' teaching has primarily focused on syntactical rules and technical diagram creation, often neglecting students' challenges in interpreting and conveying meaning through diagrams [16, 26]. By analysing UML class diagrams through a semiotic lens, educators can better understand students' conceptual knowledge, visual communication skills, and the factors influencing their diagram interpretations [29, 45]. This can help guide more effective teaching strategies that address UML class diagrams' technical and communicative aspects.

Previous research has shown that students struggle more with UML semantics than syntax [13]. A semiotic approach can uncover the sources of these difficulties by exploring how students interpret UML class diagrams' symbols, navigate cultural and contextual influences, and construct meaning through diagrams. This study extends this research by applying the six principles of semiotic analysis to examine how first-year BCIS students engage with UML class diagrams. Understanding the semiotic aspects of UML diagramming allows educators to develop more effective pedagogical strategies addressing technical and communicative elements. This approach provides insights that traditional analyses focused only on technical correctness may miss, offering a more holistic view of how students learn and apply UML class diagram concepts.

3 Research design and methods

3.1 Design

The study used a narrative approach and followed an integrated-methods research framework based on Plowright's [40] FraIM (Frameworks for an Integrated Methodology). FraIM, a mixed-methods design, does not require a specific philosophical stance at the beginning of the study. Instead, such a stance may develop as the study progresses or when interpreting the results. Data for the narrative analysis were collected through interviews, artifacts, and observations. The study focused on first-year Bachelor of Computer Information Systems (BCIS) students at a university in South Africa. First-year students represent an ideal population for investigating pedagogical UML interpretation because their developing expertise reveals fundamental semiotic processes that become unconscious in professional practice. The participants were self-selected based on their

voluntary participation in the research activities. The sample was chosen for both purposeful [46] and convenience [28, 35] reasons. It was purposeful because the students were enrolled in a module covering the studied concepts (aggregation and composition). It was also convenient because one of the researchers (the third author) was the module instructor, making access to the students easier. Of the 105 students in the class, 20 were invited to participate on a first-come, first-served basis. Ultimately, 11 students were able to take part.

The sample size of 11 participants is appropriate for this qualitative narrative study. Guest et al. [22] found that saturation often occurs within 6–12 interviews in homogeneous samples, and other authors (e.g [10, 27]), note that 5–25 participants are typical in phenomenological research. For think-aloud protocol research specifically, Ericsson and Simon [15] indicate that 8–15 participants are sufficient to identify recurring patterns. Our homogeneous group of first-year BCIS students studying the same UML concepts enabled consistent thematic saturation across the six semiotic principles. Before data collection, ethical approval was obtained from the relevant review board (Reference: UFS-HSD2023/1768).

3.2 Research methods

As part of the data collection process involving 'asking questions,' primary data were gathered through individual semi-structured interviews. These interviews took place in three phases, each designed to reveal the thought processes students followed when tackling tasks of different levels of complexity: a simple task in phase one, a moderately challenging task in phase two, and a more complex task in phase three.

Participants were given pen and paper to complete the activities, and their written responses were collected as artifacts for further analysis. Participants were asked to use the think-aloud method to gain insight into their thought processes. The interviews were audio-recorded with their consent to ensure accurate documentation of their verbal responses. In addition, the researcher observed participants' behaviours during the interviews, recording their reactions and interactions in a research diary. This combination of data sources helped to triangulate the findings and supported the overall interpretation of the study.

Phase 1 – Participants were asked if they were familiar with the diagrams shown in Fig. 1. They were then instructed to carefully examine each diagram and describe their observations based on concepts covered in the module. The diagrams were not labelled to promote independent thinking, requiring participants to identify them independently. As seen in Fig. 1, diagrams 1, 2, and 3 represent the concepts of association, aggregation, and composition, respectively. Association is a general relationship between two classes indicating a connection without implying ownership. Aggregation (hollow diamond) indicates a 'has-a' relationship where parts can exist independently of the whole.

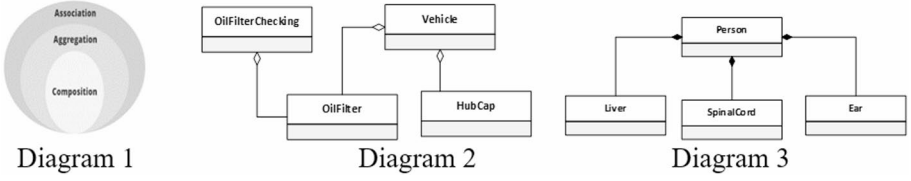


Fig. 1 Diagrams Used for Phase 1 Activities

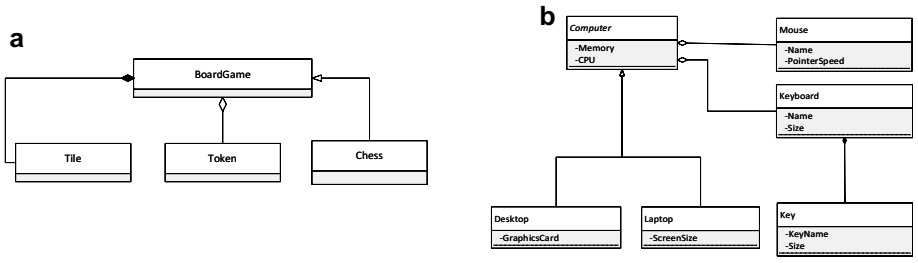


Fig. 2 **a** Boardgame class diagram, **b** Class diagram for a computer and its components

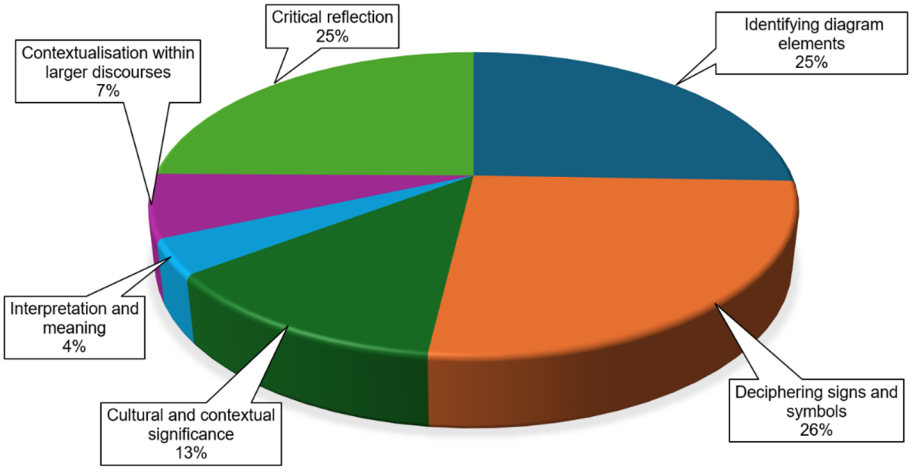


Fig. 3 Percentage of strategy occurrences per semiotic principle

Composition (filled diamond) is a 'whole–part' relationship where the part cannot exist independently of the whole [16, 26].

Phase 2 — In this phase, participants were tasked with describing the relationships depicted between the classes illustrated in Fig. 2 and to explain the class diagram shown in Fig. 3.

Phase 3 – This phase consisted of two separate tasks. For the first task, participants were asked to create a class diagram based on the following specifications:

1. A computer has private attributes, such as Memory and CPU.
2. A mouse and keyboard are components of a computer.
3. The mouse has the private attributes: Name and Pointer-Speed.
4. The keyboard has the private attributes: Name and Size.
5. A keyboard contains keys, and each has the private attributes: KeyName and Size.
6. A desktop and a laptop are types of computers.
7. The private attribute for a desktop is GraphicsCard, while the private attribute for a laptop is Screensize.

In the second task, participants were asked to explain the class diagram shown in Fig. 3. *(Note: This diagram was the same basic one the participants were expected to create in the first task of phase 3).*

The UML class diagrams deliberately focused on structural relationships (aggregation, composition, association, inheritance) while omitting behavioural elements such as methods. This design choice aligns with introductory UML pedagogy where structural

understanding precedes behavioural modelling [26]. Following the cognitive load theory, the simplified diagrams allowed students to focus on relationship semantics without additional complexity from method specifications. This approach reflects standard pedagogical practice of incremental UML introduction [16].

3.3 Data analysis

Following the methodological framework established by Creswell and Creswell [9], the audio recordings of the interviews were transcribed and thoroughly analysed. The first step in data cleaning involved addressing inconsistencies in the transcriptions, particularly due to the open-ended nature of the questions, which sometimes led to repetitive or unclear responses. To manage these issues, a fuzzy validation method was applied, allowing for subtle adjustments while maintaining the authenticity and coherence of the data [33].

Diagram quality and communicative correctness were assessed through triangulation across data sources. Structural completeness was evaluated by comparing participant diagrams against given specifications. Communicative correctness was determined by analysing whether participants' verbal explanations aligned with their visual representations and whether diagrams effectively conveyed intended relationships to the research team.

To gain a deeper understanding and prepare for coding, the researchers engaged in an immersive process, repeatedly listening to the audio recordings and carefully reviewing the transcripts. This familiarisation phase helped shape a structured coding framework aligned with the study's research questions. Once refined, the 11 verified transcripts were uploaded into NVivo software for further analysis. Codes were developed to capture the emerging strategies participants communicated based on the observed data. The names of the codes consisted of single words or simple phrases. Additionally, the names of these codes were regularly reviewed, with some being merged or renamed as necessary. During this stage, it was essential to repeatedly review the transcripts while paying close attention to the level of detail in the analysis units. Relevant text segments were systematically annotated, allowing for the identification of connections between codes and their corresponding excerpts. Throughout this iterative process, the code names were consistently refined to capture thematic patterns better, specifically emphasising the frequency and significance of recurring themes.

3.4 Reliability and validity

The validity and reliability of the study were ensured through multiple strategies. Triangulation was employed by comparing think-aloud verbalisations with participants' written diagrams, researcher observations, and artefacts such as answer scripts. This approach allowed for identifying both convergences and divergences in the participants' responses. Member checking was conducted with five participants to further enhance credibility, who reviewed and verified the researchers' interpretations of their diagram-reading strategies. Additionally, transcribed interviews were sent to all 11 participants to confirm that the transcripts accurately reflected the ideas they had shared, and all agreed that the transcripts reflected the true views they had expressed. Coding frameworks were established to analyse written diagrams and verbal data and improve inter-rater reliability. All three researchers cross-checked the interpretations made to ensure

consistency. Structured observation protocols were used to guide researcher observations systematically to mitigate subjectivity. Finally, ecological validity was considered by designing tasks that closely mirrored real-world problem-solving scenarios relevant to the participants' academic context.

4 Results and interpretation

This section presents the key findings from analysing the data collected during the 11 think-aloud sessions. The findings are organised according to the six principles of semiotic analysis outlined in Sect. 2, focusing on the top-most strategy within each principle. These strategies are presented together with the percentage distribution of their occurrence in the data in Fig. 3. All the excerpts used represent quotes from the semi-structured interviews. While many artefacts were collected, only the most note-worthy one was presented due to space limitations.

4.1 Identifying diagram elements

Analysis of the think-aloud data and the interviewer's observations revealed that the students were communicating *where to start looking* as the top-most strategy under this principle. This strategy was identified in all participants with a total of 34 occurrences.

The data reveals a range of approaches that students use to determine where to begin (where to start looking) when analysing UML class diagrams. These approaches highlight differences in the students' levels of consistency and understanding. All participants demonstrated this strategy, with some showing consistent approaches and others displaying inconsistency. For example, P1 demonstrated an inconsistent approach – In one instance, she focussed on abstract classes first:*"With the vehicle and oil filter, I think it's composition, and with oil filter and oil filter checking, it's also composition"*. In other instances, she focussed on the sub-classes first: "*I see the liver, the spinal cord and the ear which are parts of the human body"*. This lack of consistency and a misunderstanding of fundamental concepts such as aggregation and composition suggests that students may struggle to identify a clear entry point for analysing UML class diagrams.

In contrast, P2 adopted a more consistent relationship-first approach, prioritising identifying relationships between elements, such as distinguishing between aggregation and composition based on visual cues like the diamonds. In this regard, she said: "*I check the diagram — What is it like? In this case I can see that it's an open diamond, so it's an aggregation relationship"*. When asked again to remind the interviewers how she attempted the activities, she said: "*I went to the diagram and checked it, is this composition or aggregation? So, I noticed that this one is an aggregation relationship. I notice it's aggregation because this is an open diamond, right? So, if it was a closed diamond, then it was going to be composition"*. This approach highlights the importance of analysing the connections between elements in the diagram before identifying the classes themselves.

Other students, such as P3 and P5, adopted a top-down approach, focusing first on the base or abstract class before examining the relationships and sub-classes. For instance, the first statement uttered by P3 when starting with Phase 2 Activity was: "*Here the abstract class or the main class is board game … there is a chess class that inherits from the board game class"*. P5, in particular, emphasised the importance of visual elements such as arrows and layout as he started Phase 1 Activity 1 by saying: "*I usually start by checking whether the lines are shaded inside, especially the arrows… I know how my mind*

works, it wants to visually see something first before going to the instructions. Then when I read it, it is going to be simple for me … the vehicle has the oil filter and then the hub cap and then the oil filter checking".

Visual elements also significantly influenced students' initial focus. P9 and P11 noted that size and bold text often draw their attention first when analysing UML class diagrams. P9 said: "*Yes, the diagram is bigger. It's the biggest thing on the page. Alright, so it caught my attention*". This means that students look at something that catches their attention before everything else. P11 said: "*If there was maybe a bold word, I'm sure I would first look at that bold word without even looking at something else, it catches the eye*". This is further evidence that students look at the UML class diagram by virtue of the space it occupies or its size, which is in agreement with Störrle [47].

4.2 Deciphering signs and symbols

Analysis of both the think-aloud data and the interviewers' observations revealed that the students were communicating *interpreting symbols* as the top-most strategy under this principle. This strategy was identified in all participants with a total of 35 occurrences.

The most prominent approach observed was for students to carefully interpret UML class diagram symbols before proceeding with their analysis. Notably, students exercised caution in understanding these symbols before progressing with their evaluation. This careful approach was evident in students' recognition of key UML concepts such as aggregation, composition, and inheritance. For instance, P2 stated: *"I said this is aggregation because of the open diamond … like I said, immediately when I see the diamond, I know it's either aggregation or composition. So, now I know aggregation is open and composition is the shaded one"*. This demonstrates that students relied on visual cues, such as the open and shaded diamond to differentiate between relevant concepts.

Beyond aggregation and composition, the *Phase 2 Activity* also included an inheritance relationship (see Fig. 2). Some students recognised it, while others could not. For example, P2 correctly identified inheritance: *"Yes, it represents inheritance … chess inherits from a board game … the desktop and laptop both inherit from the computer and this means they are subclasses"*. P5 further demonstrated awareness of UML notation when working on the *Phase 2 Activity*, saying: *"I see an arrow here that is different from the previous ones [for all Phase 1 Activities] … that symbol says there is inheritance here"*. In *Phase 3 — Activity 2*, P5 further reinforced this understanding by stating: *"I just start by looking at the base class in this diagram"*.

However, not all students correctly interpreted all the UML diagram symbols. Some students struggled to identify certain relationships or confused the meaning of specific notations. For instance, in *Phase 2 Activity*, P9 misidentified the meaning of the diamond symbol, stating: *"The arrowhead of the relationship line indicated to me that it is generalisation"*. Instead of recognising the diamond as representing aggregation or composition, the student associated it with generalisation.

4.3 Cultural and contextual significance

Analysis of both the think-aloud data and the interviewer's observations revealed that the students were communicating *context* as the top-most strategy under this principle. This strategy was identified in nine participants with a total of 17 occurrences.

The findings highlight the importance of *context* in students' understanding and interpretation of UML class diagrams. It was observed that students without the requisite background struggled to grasp certain elements of the diagram(s). Emphasising the role of context, P7 remarked: *"Can I ask, what is a hub cap?"* This question arose during *Phase 1 — Activity 1* and underscores the significance of using familiar terms for UML classes. When asked later: *"For you to be able to explain this diagram, was it important that you should understand what each of these names for the classes means?"*, P7 responded: *"Yes, it is important for me to know because without knowing what it is, I don't know whether or not it is that important to the abstract/main class. I need to know what it is or what it does"*.

While understanding the names of classes is crucial, P7 was still able to proceed with the analysis, making some correct interpretations: *"The hub cap, I don't know what it is, but it cannot exist without a vehicle. This is understandable in this diagram"*. Similarly, P1 shared: *"A vehicle sometimes contains an oil filter, and I don't know what a hub cap is, but then I just feel like it's also part of the vehicle"*. This suggests that while some students may not fully understand specific terms, they often use contextual reasoning to make connections. P9 added: *"When I know the name, it helps me understand better"*. This reinforces the idea that context plays a significant role in students' ability to interpret UML class diagrams.

4.4 Interpretation and meaning

Analysis of both the think-aloud data and the interviewer's observations revealed that the students were communicating *confusing terms* as the top-most strategy under this principle. This strategy was identified in three participants with a total of five occurrences.

The most significant challenge in interpreting UML class diagrams was the confusion between related terms and concepts. The theme of confusing terms and concepts emerged as some participants experienced difficulty differentiating between related but distinct concepts, such as aggregation and composition. However, they still clearly understood the symbols associated with these relationships. The confusion reflects the challenge of accurately applying theoretical knowledge to practical scenarios. For instance, P7 stated: *"The vehicle has an oil filter and an oil filter cannot exist without a vehicle and oil filter must be checked. To check the oil filter, there must be an oil filter. So, the changing of the oil filter cannot exist without the oil filter"*.

While P7 demonstrated an understanding of the dependencies between the vehicle and oil filter, there was confusion between the types of relationships. When asked if there were symbols that influenced his understanding, P7 responded: *"I have forgotten the names, composition or what, but I understand those symbols and how they work. But the problem is to tell whether this is composition or aggregation or something else"*. This statement illustrates that while P7 could identify and understand the visual symbols used in the UML class diagram, distinguishing between aggregation and composition remained challenging. P7 also produced an artifact in which he misidentified the shaded diamond as representing aggregation (Aggr) and the unshaded diamond as representing composition (Comp) (see Fig. 4). This further confirms the confusion between the terms, despite the participant's familiarity with the symbols.

Similar terminological confusion was observed in other participants. P5 expressed similar confusion during Phase 3 Activity 2: *"There is a relationship between a keyboard*

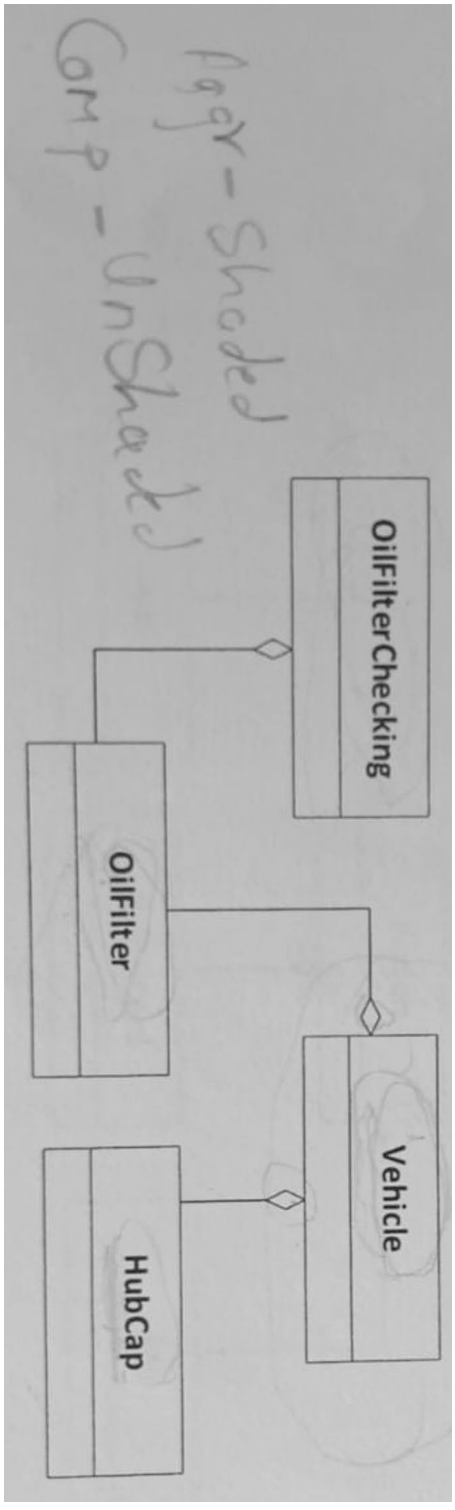


Fig. 4 P7's artefact for Phase 1 Activity 1

and a key. This relationship is composition or aggregation, I forgot. And then what I know is that a shaded diamond shape says a keyboard cannot live without a key. So, it seems like a key is contained inside the keyboard". P5 also struggled to recall the correct terminology, despite articulating the relationship well. When asked why he had difficulty

remembering the concepts, he explained: *"For me, I am not a theoretical person, I don't like theory, hence why it is a problem, but I can understand things [theoretical aspects]".* This response highlights that while P5 understood the symbolic relationships and could apply them contextually, remembering and distinguishing the correct theoretical terms posed a significant challenge.

The aggregation-composition confusion observed reflects the dual learning challenge these relationships present. Unlike simple association (plain line), both aggregation and composition use diamond notation, requiring students to distinguish semantic concepts through subtle visual differences. This creates challenges at two levels: students must master both the semiotic distinction (hollow vs. filled diamonds) and the conceptual distinction (independent vs. dependent existence). From a semiotic standpoint, Siau and Tian [45] show that more iconic visual notations in UML, such as hollow or filled diamonds, lead to more accurate interpretation and fewer unintended connotations among learners. This supports the idea that maintaining clear visual metaphors in class diagrams helps novice students bridge the gap between visual form and conceptual meaning.

4.5 Contextualisation within larger discourses

Analysis of the think-aloud data and the interviewer's observations revealed that the participants communicated *real work terminology* as the top-most strategy under this principle. This strategy was identified in seven participants with a total of nine occurrences.

It was key to note that students resonated more with activities that incorporated names or terms for classes using *real-world terminology/language*. For example, P2 responded: "*Yes*" when asked as follows: "*So, are you trying to say using the real-world examples help because now if you know how a computer works then it's easier for you to make links to the concepts or terms in a scenario?"* This also surfaced with P4 because when he attempted *Phase 1 — Activity 2*, he straightaway said: "*This person has a spinal cord, the ear, and the liver*". One could develop an impression that the student understood the relevant concepts incorporated in the UML class diagram. However, he seemed to have used the context of being familiar with a person, as the interviewers observed that he struggled to recall the necessary concepts and relationships between the classes [3, 11, 36]. For this activity, he ultimately said: "*Can I skip this one?".* He was allowed to do so, but he later could recall some concepts and relationships due to lengthy prompts that the interviewers engaged in to guide him. Returning to the same activity, he said: "*The ear cannot exist without the person. Yes, the ear and the spinal cord cannot exist without the person, and the liver cannot exist without the person*".

Irrespective of the above experiences, students expressed differing views on using familiar real-world terminology in UML class diagrams. P10 argued that understanding the specific names was not crucial, as she could still make connections based on the relationships and symbols. She said: *"So it doesn't really matter for me to understand the diagram … it shows that it's coming from a vehicle. I know it's part of the vehicle"*. P10 further explained that even if the names were unfamiliar, she could rely on the diagram's structure and symbols to interpret the relationships: *"If you forget the name, you should look at the arrows… If you can't remember that this is called aggregation, then you should at least know that it's something like a diamond without being stated"*. P9 shared a similar perspective, acknowledging that although familiar terms could make the task easier,

they were not essential. She responded: "*Yes*" to this statement that was put to her by one of the interviewers: *"But according to you, you would relate better if you know the components, if you know the terms?"*. In contrast, P11 emphasised that familiarity with the class names could help clarify confusion when interpreting the diagram. She stated: *"If it's another concept I don't know, wherever I get confused, there's nothing that will give me a hint because I don't know these things"*.

4.6 Critical reflection

Analysis of both the think-aloud data and the interviewer's observations revealed that the students were communicating *name of the UML class diagram* as the top-most strategy under this principle. This strategy was identified in all participants with a total of 33 occurrences.

The strongest reflection occurred when students were asked to name UML class diagrams, revealing their conceptual understanding. Participants provided diverse responses when asked to *name the UML class diagrams* for Phase 1 Activities 1 and 2. Ideally, students were expected to incorporate UML concepts such as aggregation, composition, or class diagrams in their naming (see Figs. 2 and 3). However, their responses varied significantly, reflecting their interpretations of the class diagrams.

Some participants named the diagrams based on their perceived subject matter. P1 suggested *"the vehicle system"* and *"the human body system or the human parts"*. Similarly, P2 associated the diagrams with specific functionalities, stating, *"I see a vehicle and an oil filter checking, so this is supposed to be about a car and how it operates within an engine"*, ultimately naming it "*Car Engine*". P3 and P5 followed a similar pattern, suggesting names like *"Internal parts of a car"*, *"The parts of a person"*, *"Vehicle something"*, and *"Living person"*.

Others took a more UML-oriented approach. For example, P4 named them *"Vehicle class diagram"* and *"Person class diagram"*. P7 and P8 maintained this UML focus but added context, using: *"Class diagram for oil filter checking"* and *"Motor Vehicle"*. P9 initially labelled both diagrams as *"Class Diagram"* but later reconsidered, asking: *"Can I change my first one to class diagram as well?"*. This suggests she was still reflecting on the most appropriate title. Notably, she first wrote *"Use Case Diagram"* before deciding to align both names with class diagrams. P10 asked the interviewers whether she could use a phrase or sentence for naming, indicating uncertainty about how to label the diagrams. She ultimately suggested: *"The diagram that shows the interaction of vehicle components"* and *"The diagram that shows (composition and person are part of the name"*. Her question highlights the need for explicit instruction on naming UML class diagrams, as seen in the problem specifications.

Beyond naming, participants demonstrated verification behaviors indicating critical reflection. P1 systematically checked completeness by marking specifications. P7 showed metacognitive awareness: "*I understand those symbols and how they work. But the problem is to tell whether this is composition or aggregation*". P9 demonstrated self-correction by revising her diagram classification.Overall, the variation in responses indicates that participants engaged in critical reflection [3, 11, 36] when naming the class diagrams. This suggests that students rely on real-world associations, while others aim for UML-appropriate terminology, highlighting differences in how they conceptualise the diagrams.

5 Discussion

This section discusses the findings presented in the previous section, analysing them through the lens of the six semiotic principles and situating them within existing literature. The discussion illuminates the communicative aspects of UML class diagrams and their implications for teaching and learning, focusing on the top-most strategy identified for each principle.

5.1 Identifying diagram elements

Under this principle, the findings identified *'where to start looking'* as the dominant strategy for students when first encountering UML class diagrams. Students who consistently applied a systematic approach demonstrated a clearer understanding and more accurate interpretations, supporting Larman's [26] assertion that structured analytical frameworks enhance comprehension of complex diagrams.

What emerged as particularly significant was the diversity in how students determine their initial focus. Some students, such as P2, prioritised identifying relationships between elements, focusing on visual cues such as diamond shapes to distinguish between 'aggregation' and 'composition'. Others, such as P3 and P5, employed a hierarchical approach by first identifying the main or abstract class before examining relationships. Both these systematic approaches led to more accurate UML class diagram interpretation compared to inconsistent approaches.

Significantly, visual prominence strongly influenced initial focus, with students such as P9 and P11 reporting that size, bold text, and spatial arrangement directed their attention — a finding consistent with Störrle's [47] research on perceptual factors in diagram interpretation. This highlights how the UML class diagram design itself communicates implicit hierarchies of importance beyond the formal notation.

The inconsistent approaches observed in some students reflect a lack of systematic analytical strategies, which impeded their ability to identify relationships and understand structural elements correctly. This suggests that developing consistent analytical frameworks is more crucial than using a specific approach, provided it is applied systematically across different class diagrams. These findings emphasise the need for UML class diagram education to focus on notation rules and developing structured approaches to UML class diagram analysis. This aligns with the cognitive load theory research [48], which demonstrates that systematic approaches reduce extraneous cognitive load, enabling students to allocate more mental resources to conceptual understanding rather than procedural uncertainty.

5.2 Deciphering signs and symbols

Under this principle, '*interpreting symbols*' emerged as the dominant strategy for students when working with UML class diagrams. Students demonstrated varying proficiency levels in interpreting UML class diagram notation, with most correctly distinguishing between aggregation and composition relationships based on visual cues such as open versus shaded diamonds. This reflects what Barthes [3] describes as the denotative function of symbols — their direct, representational meaning within the UML class diagram notation system.

However, a significant pattern emerged where students could recognise symbols visually but struggled with their formal terminology, particularly with inheritance

relationships. This disconnect between visual recognition and conceptual labelling reveals a gap between the signifier (the symbol) and the signified (the concept it represents), as articulated in de Saussure's [11] semiotic theory. For example, P3 correctly understood that chess was a type of board game, but could not name this as an '*inheritance*' relationship, despite recognising the arrow notation.

These misinterpretations and terminological difficulties align with what Dobing and Parsons [13] identified as common misconceptions in understanding UML class diagrams. Their research suggests that students often grasp the visual syntax before mastering the semantic meaning — recognising what a symbol looks like before fully comprehending what it represents conceptually. This reflects Eco's [14] concept of '*aberrant decoding*', where interpretations diverge from intended meanings due to different levels of familiarity with the code system.

The findings suggest that UML class diagram education should strengthen the connections between visual notation and conceptual understanding, rather than treating them as separate pedagogical components. Moody's [29] research on visual notations supports this approach, suggesting that effective notational systems must consider both the cognitive and perceptual dimensions of interpretation. Pedagogical strategies might include exercises that explicitly require students to verbalise the meaning of symbols in various contexts, strengthening the connection between visual recognition and conceptual understanding. This recommendation is supported by dual coding theory [32], which suggests that learning is enhanced when visual and verbal information are processed simultaneously and connected explicitly.

5.3 Cultural and contextual significance

The findings revealed '*context*' as the most influential factor in how students interpret UML class diagrams influenced by cultural and contextual significance. This indicates that familiar terminology significantly affects students' ability to comprehend diagram elements and their relationships. When confronted with unfamiliar terms like 'hub cap', students such as P7 explicitly questioned their meaning, recognising that understanding class names was crucial for accurate interpretation. This importance of contextual familiarity supports Eco's [14] semiotic theory, which posits that interpretation is deeply rooted in prior experiences and knowledge frameworks. Students lacking contextual understanding of terminology must exert additional 'cognitive effort' to make sense of the diagram, creating potential barriers to comprehension.

Interestingly, several students demonstrated contextual reasoning to bridge knowledge gaps, as when P1 inferred that a 'hub cap' must be 'part of the vehicle' despite not knowing its specific function. This reflects what Peirce [36] describes as abductive reasoning — a form of logical inference that uses available information to make the most likely explanation. Students actively constructed meaning by connecting unfamiliar elements to familiar contexts, demonstrating that interpretation involves active meaning-making rather than passive reception.

These findings suggest that UML class diagram pedagogy should address contextual factors using familiar terminology that resonates with students' existing knowledge base, particularly in the introductory stages. However, educators should also systematically expand students' exposure to diverse terminologies and UML class diagram arrangements to develop their ability to interpret unfamiliar contexts. The balance between

leveraging familiar contexts and stretching students' interpretive abilities presents a key pedagogical challenge in UML class diagram instruction. This finding extends Vygotsky's [51] zone of proximal development concept to technical diagramming: learning is most effective when familiar contexts scaffold students' grappling with abstract notation, enabling progression from what they can do alone to what they can accomplish with support.

5.4 Interpretation and meaning

The findings identified '*confusing terms*' as the most significant challenge in students' interpretation of UML class diagrams. While students often grasped the visual representation of relationships, they frequently struggled to apply the correct terminology, particularly when distinguishing between related concepts like aggregation and composition.

This conceptual confusion reflects the arbitrary relationship between signs and their meanings that de Saussure [11] describes as fundamental to semiotic systems. Concepts such as 'aggregation' and 'composition' gain their meaning through convention rather than inherent qualities, requiring deliberate learning of these concepts. The challenge is amplified when concepts are closely related but conceptually distinct, as P7 demonstrated when stating that he had forgotten the relevant names, such as composition and aggregation, but he understood the associated symbols and how they work. This finding is particularly noteworthy because students often correctly understood the underlying concepts despite lacking terminology. For example, P5 accurately described that a shaded diamond shape meant that a keyboard could live without a key, but could not confidently label that as an '*composition*' relationship. This suggests students develop an intuitive functional understanding of relationships (comprehension and application) before mastering formal terminology, consistent with Bloom's [4] hierarchy. It also aligns with Chi et al. [8], who found novices typically understand and apply concepts before they can declaratively articulate them precisely.

5.5 Contextualisation within larger discourses

The findings revealed that students' engagement with '*real-world terminology*' was the dominant strategy for contextualising UML class diagrams within broader discourses. The varying perspectives on real-world terminology reflect different approaches to Barthes' [3] 'anchoring' using textual elements to fix the meaning of visual signs. Some students relied on familiar terminology, while others focused on structural relationships. This difference highlights de Saussure's [11] distinction between 'paradigmatic' understanding (focusing on relationships) and 'syntagmatic' understanding (focusing on elements in context). Students who interpreted UML class diagrams without familiar terminology showed a stronger paradigmatic understanding of their structure. This finding is significant because it reveals how students bridge the gap between abstract UML class diagram notation and their existing knowledge frameworks. For example, P2 explicitly confirmed that using real-world examples is helpful because understanding how a computer works makes connecting concepts to a given scenario easier. Similarly, P4's immediate recognition of familiar body parts *("This person has a spinal cord, the ear, and the liver"*) demonstrated how real-world knowledge provides an entry point for technical interpretation.

At the same time, some students demonstrated greater abstraction abilities, with students such as P10 noting that understanding the name of the UML class diagram is not so crucial as when the name is forgotten, one should consider interpreting the associated arrows, indicating a focus on structural relationships over terminology. This contrast in approaches reveals different levels of progress toward abstract diagrammatic thinking.

These findings suggest that UML class diagram pedagogy should balance both approaches — introducing concepts with real-world examples while gradually focusing on structural relationships. This balance would support students who learn through concrete examples and those who prefer abstract structural thinking in agreement with Larman's [26] findings. This dual approach also reflects constructivist learning theory [39], where students build understanding by connecting new abstract concepts to existing concrete knowledge structures.

5.6 Critical reflection

The findings identified the *'naming of UML class diagrams'* as the strongest indicator of students' critical reflection abilities. The variety in UML class diagram naming approaches reflected differences in conceptual framing — some focused on the subject matter (e.g., '*human body system*'), while others prioritised technical classification (e.g., '*class diagram*'). This variation illustrates Eco's [14] idea of different '*codes*' or interpretive frameworks students use to make sense of UML class diagrams.

The naming choices particularly revealed students' conceptual understanding. Those who selected concrete, domain-specific names such as "*the vehicle system*" or "*Car Engine*" demonstrated a focus on the diagram's subject matter rather than its structural properties. This naming approach suggests these students prioritised real-world referents over the diagram's technical classification. In contrast, students who chose names such as "*Vehicle class diagram*" showed greater awareness of the class diagram's place within UML's formal classification system. This finding is significant because naming requires students to prioritise certain aspects of the UML class diagram over others, revealing their dominant interpretive framework. P9's reconsidering of her initial label from "*Use Case Diagram*" to "*Class Diagram*" demonstrates active critical reflection and conceptual refinement. Similarly, P10's question about whether she could use a phrase instead of a single term shows awareness of naming conventions and their implications.

These findings suggest that UML class diagram pedagogy should encourage critical reflection through activities that prompt students to assess the diagram's completeness, consider alternative interpretations, and question assumptions. Specifically, asking students to name UML class diagrams and justify their naming choices could be a valuable pedagogical tool for revealing and developing their conceptual understanding. Such practices could deepen conceptual understanding and improve students' diagram communication skills in agreement with the findings of Fowler [16]. This metacognitive approach is supported by research on expert-novice differences, where Glaser and Chi [18] found that experts consistently employ systematic self-monitoring and evaluative strategies far more than novices.

5.7 Pedagogical implications

The focused analysis of the dominant strategies employed by students across the six semiotic principles reveals significant implications for UML class diagram pedagogy:

- **Systematic approaches to initial analysis**: The study findings on 'where to start looking' suggest that educators should explicitly teach consistent analytical frameworks for UML class diagram interpretation. Whether students adopt a relationship-focused or top-down approach matters less than maintaining consistency across different diagrams. Teaching materials should demonstrate these systematic approaches rather than assuming students will develop them independently.
- **Connecting visual notation with conceptual understanding**: Students' challenges in 'interpreting symbols' highlight the need for educational approaches that strengthen connections between visual recognition and conceptual understanding. Exercises should require students to articulate the meaning of symbols in various contexts, reinforcing the link between notation and concept. This recommendation aligns with empirical research by Siau and Tian [45], who demonstrated that UML notations with greater visual resemblance to their referents (iconic notations) are interpreted more accurately than purely arbitrary (symbolic) notations.
- **Contextual scaffolding for interpretation**: The importance of 'context' in students' understanding suggests that educators should use familiar terminology that resonates with students' existing knowledge, particularly in introductory stages. As students progress, exposure to diverse terminologies can develop their ability to interpret unfamiliar contexts.
- **Addressing terminological precision**: The 'confusing terms' challenge indicates a need for focused learning activities that help students distinguish between related concepts like aggregation and composition. Comparative exercises that highlight the differences between similar relationships can strengthen terminological precision.
- **Balancing concrete examples with abstract structures**: Students' varying reliance on 'real-world terminology' suggests UML education should progress from concrete, familiar examples toward more abstract structural understanding, supporting both students who learn through real-world analogies and those who prefer abstract thinking.
- **Using naming as a diagnostic and developmental tool**: The varying approaches to 'naming UML diagrams" can be leveraged to assess conceptual understanding and as a learning activity. Asking students to name diagrams and justify their choices reveals their interpretive frameworks and promotes critical reflection.
- Progressive complexity in UML diagram design: The deliberate omission of methods from the study diagrams highlights the need for scaffolded UML diagram instruction. While a simplified structural focus allowed students to concentrate on relationship semantics without cognitive overload, it may reinforce incomplete mental models of object-oriented design. UML diagram instruction should begin with structure-focused diagrams to support relationship understanding. As learners gain confidence, methods for encapsulating behaviour can be gradually introduced. This progression should ultimately lead to comprehensive modelling that integrates both structural and behavioural elements. When methods are introduced, they should be presented semiotically, explaining not just syntax but the communicative significance of visibility modifiers and parameter specifications.

- Teaching explicit verification strategies: Teaching students explicit verification strategies, such as systematic completeness checking and accuracy validation, can enhance their critical reflection skills and improve diagram quality.

By addressing these dimensions based on students' actual semiotic engagement with UML class diagrams, educators can develop more effective approaches that bridge technical notation with meaningful communication of design concepts.

6 Conclusion

This study applied semiotic theory to analyse how first-year BCIS students communicate through and interpret UML class diagrams. By examining the six principles of semiotic analysis and identifying the strongest strategy within each, valuable insights into students' engagement with these diagrams as communicative tools were gained. By examining students in educational rather than professional contexts, this study captures the fundamental semiotic processes underlying UML diagram interpretation before they become automated through workplace experience.

The study findings reveal that UML class diagrams function as complex semiotic systems through which students convey and construct meaning. For each semiotic principle, a dominant strategy that characterised students' engagement was identified:

- *Where to start looking*: Students' initial analytical approaches significantly influenced their UML class diagram interpretation, with consistent methods yielding better understanding.
- *Interpreting symbols*: Students demonstrated proficiency in visual recognition of UML class diagram notation, but often struggled with terminological precision.
- *Context*: Familiar terminology substantially affected comprehension of UML class diagrams, with students using contextual reasoning to bridge knowledge gaps.
- *Confusing terms*: Students frequently distinguished between related concepts visually but struggled to apply correct terminology associated with UML class diagrams.
- *Real-world terminology*: Students showed varying reliance on familiar terminology used with UML class diagrams, reflecting different stages in developing abstract diagrammatic thinking.
- *Naming UML class diagrams*: The diverse approaches to naming revealed students' conceptual understanding and interpretive frameworks.

These findings highlight key challenges in UML class diagram pedagogy, particularly the gap between visual recognition and conceptual understanding, the role of context in interpretation, and the development of abstract thinking from concrete examples. The semiotic analysis framework proved valuable for understanding students' engagement with UML class diagrams beyond surface-level correctness, revealing the deeper communicative processes at work.

This research contributes to both pedagogical practice and theoretical understanding of visual communication in technical contexts. It offers specific pedagogical recommendations for each semiotic dimension and demonstrates the value of semiotic theory in analysing technical diagrams as communicative artifacts embedded in broader educational and professional discourses.

7 Limitations and future research

This study offers valuable insights into how students engage with UML class diagrams as semiotic systems, but several limitations should be noted. First, focusing on first-year BCIS students at a single South African university limits generalisability to other educational contexts, cultures, and expertise levels. The local curriculum and teaching methods may have influenced students' understanding differently. Second, the small sample size (11 participants) and self-selection bias (with participants likely more interested or confident in UML class diagrams) limit statistical generalisability. Third, the focus on structural UML elements (classes, attributes, relationships) while excluding behavioural elements (methods, operations) represents a limitation in the study's scope. While this design choice was pedagogically motivated and appropriate for investigating the semiotic interpretation of relationship types, it means the findings may not fully generalise to complete UML class diagram interpretation that includes behavioural modelling. The focus on pedagogical contexts limits direct generalisability to professional UML practice, where established expertise and workplace conventions may influence semiotic interpretation differently. Fourth, the think-aloud protocol, while insightful, may have influenced how students approached tasks, as verbalising thoughts could have altered natural engagement. Fifth, being cross-sectional, the study provides only a snapshot of students' understanding at one point in time without tracking how their engagement might evolve.

These limitations suggest several directions for future research. Comparative studies across institutions, educational levels, and cultural contexts could improve generalisability and reveal how different approaches affect students' engagement with UML class diagrams. Longitudinal studies could track students' evolving understanding, identifying key stages and learning thresholds. Expanding the analysis to include various UML diagram types would offer a broader understanding of semiotic processes and how students transfer interpretive strategies across types. Experimental studies could test the effectiveness of pedagogical interventions designed explicitly around the six dominant strategies identified in this study: systematic initial approaches (*where to start looking*), connecting visual notation with conceptual understanding (*interpreting symbols*), contextual scaffolding (*context*), terminological precision (*confusing terms*), balancing concrete examples with abstract structures (*real world terminology*), and using naming as a developmental tool (*name of the UML class diagram*). Eye-tracking studies could complement the think-aloud protocols by offering objective insights into how students visually process UML class diagrams, revealing unconscious patterns and their correlation with comprehension. Moreover, research comparing novices and experts could identify key transitions in developing professional competence with UML class diagrams, bridging educational and professional contexts. Future research in these areas can further deepen our understanding of UML class diagrams as communicative tools in academic and professional settings.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1007/s44217-025-00817-8.

Supplementary Material 1

Author contributions
PJK: Conceptualisation, Methodology, Formal analysis, Writing - original draft, Writing - review & editing.RCF: Conceptualisation, Investigation, Formal analysis, Writing - review & editing.TSN: Data collection, Investigation, Writing - review & editing.

Funding
The authors did not receive funding support from any organisation for the submitted work.

Data availability
The anonymised interview transcripts and student artifacts collected during this study are available from the corresponding author upon reasonable request.

Declarations

Competing interests
The authors declare no competing interests.

Received: 18 April 2025 / Accepted: 29 August 2025
Published online: 16 September 2025

References

1. Abrejo S, Baig A, Ali A, Rahman M, Khoso A. Hints-based approach for UML class diagrams. Int J Comput Sci Netw Secur. 2023;23(7):9–15. https://doi.org/10.22937/IJCSNS.2023.23.7.2.
2. Ambler SW. The elements of UML 2.0 style. Cambridge University Press; 2005.
3. Barthes R. Elements of semiology volume 4 of cape editions. Farrar, Straus and Giroux; 1968.
4. Bloom BS. Taxonomy of educational objectives, handbook 1 cognitive domain. New York: Longman; 1984.
5. Booch G, Rumbaugh J, Jacobson I. The unified modeling Language user guide. Addison-Wesley Professional; 2005.
6. Boustedt J. Students' different Understandings of class diagrams. Comput Sci Educ. 2012;22(1):29–62. https://doi.org/10.1080/08993408.2012.665210.
7. Chandler D. Semiotics: the basics. Routledge; 2022.
8. Chi MTH, Feltovich PJ, Glaser R. Categorization and representation of physics problems by experts and novices. Cogn Sci. 1981;5(2):121–52. https://doi.org/10.1207/s15516709cog0502_2
9. Creswell JW, Creswell JD. Research design: qualitative, quantitative, and mixed methods approaches. 5th ed. Thousand Oaks: Sage; 2017.
10. Creswell JW, Poth CN. Qualitative research inquiry and design: choosing among five approaches. 5th ed. Thousand Oaks: Sage; 2025.
11. de Saussure F. *Course in General Linguistics*. Translated by Wade Baskin. Edited by Perry Meisel and Haun Saussy. Columbia University Press; 2011.
12. Dennis A, Wixom B, Tegarden D. *Systems Analysis and Design: An Object-Oriented Approach with UML*, 6th Edition, Wiley; 2021.
13. Dobing B, Parsons J. How UML is used. Commun ACM. 2006;49(5):109–13. https://doi.org/10.1145/1125944.112594.
14. Eco U. A theory of semiotics. Indiana University Press; 1979.
15. Ericsson KA, Simon HA. Protocol analysis: verbal reports as data. Cambridge, Massachusetts: The MIT Press; 1993. Rev. ed.
16. Fowler M. UML distilled: A brief guide to the standard object modeling Language. Addison-Wesley Education Publishers Inc; 2004.
17. France R, Rumpe B. Model-driven Development of Complex Software: A Research Roadmap, *Future of Software Engineering (FOSE '07)*, Minneapolis, MN, USA, 2007, pp. 37–54. https://doi.org/10.1109/FOSE.2007.14
18. Glaser R, Chi TH. Overview. In: Chi TH, Glaser R, Farr MJ, editors. The nature of expertise. Hillsdale, NJ: Lawrence Erlbaum Associates; 1988. pp. 15–28.
19. Gogolla M, Kobryn C, editors. UML 2001—The unified modeling language: modeling languages, concepts, and tools. Springer; 2001.
20. Górski T. The 1 + 5 architectural views model in designing blockchain and IT system integration solutions. Symmetry. 2021;13(11):Article2000. https://doi.org/10.3390/sym13112000.
21. Guest G, Bunce A, Johnson L. How many interviews are enough?? An experiment with data saturation and variability. Field Methods. 2006;18(1):59–82. https://doi.org/10.1177/1525822X05279903.
22. Khomokhoana P, Nkalai TS. (2024). Cognitive processes used by students in understanding Aggregation and Composition in UML: A case of first-year BCIS students. In *Proceedings of the 53rd Annual Conference of the Southern African Computer Lecturers' Association (SACLA 2024)*, University Way, Summerstrand Gqeberha, South Africa, 17–19 July 2024 (pp. 116–131). ISBN: 978-1-928472-67-4 (Electronic).
23. Knopfel A, Grone B, Tabeling P. Fundamental modeling concepts: effective communication of IT systems. Hoboken, New Jersey: John Wiley & Sons, Inc.; 2006.
24. Kruchten PB. The 4 + 1 view model of architecture. IEEE Softw.; 1995; *12*(6):42—50. https://doi.org/10.1109/52.469759
25. Górski, T. (2020). Verification of Architectural Views Model 1+5 Applicability. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds) Computer Aided Systems Theory – EUROCAST 2019. EUROCAST 2019. Lecture Notes in Computer Science, vol 12013. Springer, Cham. https://doi.org/10.1007/978-3-030-45093-9_60
26. Larman C. Applying UML and patterns: an introduction to Object-Oriented analysis and design and iterative development. Pearson; 2004.
27. Mason M. Sample size and saturation in phd studies using qualitative interviews. Forum: Qualitative Social Res. 2010;11(3):Article8. http://nbn-resolving.de/urn:nbn:de:0114-fqs100387.

28. Mishra SB, Alok S. Handbook of research methodology: A compendium for scholars & researchers. Educreation Publishing; 2017.
29. Moody DL. The physics of notations: toward a scientific basis for constructing visual notations in software engineering. IEEE Trans Software Eng. 2009;35(6):756–79. https://doi.org/10.1109/TSE.2009.67.
30. Munroe R. Thing explainer: complicated stuff in simple words. Boston: Houghton Mifflin Harcourt; 2015.
31. Nadin M. Signs and systems, in signs and systems. A semiotic introduction to systems design. Cambridge: Cambridge University Press; 1997.
32. Paivio A. Mental representations: A dual coding approach. New York: Oxford University Press; 1986.
33. Parcell ES, Rafferty KA. Interviews, recording and transcribing. The SAGE encyclopedia of communication research methods. SAGE Publications, Inc; 2017. https://doi.org/10.4135/9781483381411.n275.
34. Parker A. (2024). UML Notation Tutorial: UML Diagram Symbol with Examples. Guru99. Retrieved 20 March 2025, from https://www.guru99.com/uml-notation-symbol.html
35. Patton MQ. Qualitative research & evaluation methods: integrating theory and practice. 4th ed. Thousand Oaks: SAGE; 2015.
36. Peirce CS. In: Hartshorne C, Weiss P, Burks AW, editors. Collected papers of Charles Sanders Peirce. Cambridge, Massachusetts: Belknap Press of Harvard University; 1960.
37. Peirce CS. In: Hoopes J, editor. Peirce on signs: writings on semiotic by Charles Sanders Peirce. University of North Carolina Press; 1991. http://www.jstor.org/stable/10.5149/9781469616810_hoopes.
38. Pender T. UML bible. New York: John Wiley & Sons, Inc.; 2003.
39. Piaget J. The role of action in the development of thinking. In: Gruber HE, Vonèche JJ, editors. The essential Piaget. Basic Books; 1977. pp. 17–42.
40. Plowright D. Using mixed methods: frameworks for an integrated methodology. Sage; 2011.
41. Roff TR. UML: A beginner's guide. New York: McGraw-Hill Companies, Inc.; 2003.
42. Satzinger JW, Jackson RB, Burd SD. Systems analysis and design: in a changing world. 7 ed. Ed. Cengage Learning; 2016. International.
43. Selic B. The pragmatics of model-driven development. IEEE Softw. 2003;20(5):19–25.
44. Shmallo R, Shrot T. Constructive use of errors in teaching the UML class diagram in an IS engineering course. J Inform Syst Educ. 2020;31(4):282–93. http://jise.org/Volume31/n4/JISEv31n4p282.html.
45. Siau K, Tian Y. A semiotic analysis of unified modeling Language graphical notations. Requirements Eng. 2009;14(1):15–26. https://doi.org/10.1007/s00766-008-0071-7.
46. Singh DP, Ahmed N, Gupta N. (2021). *Business Research Method And Project Work*. SBPD Publications. Retrieved 09 January 2025, from https://books.google.co.za/books?id=4-RSEAAAQBAJ
47. Störrle H. On the impact of size to the Understanding of UML diagrams. Softw Syst Model. 2016. https://doi.org/10.1007/s10270-016-0529-x.
48. Sweller J, Ayres P, Kalyuga S. Cognitive load theory. Springer; 2011. https://doi.org/10.1007/978-1-4419-8126-4.
49. Tilley S. *Systems Analysis and Design*. 13th Edition. Cengage Learning, Ohio; 2025.
50. Tyrer C. The voice, text, and the visual as semiotic companions: an analysis of the materiality and meaning potential of multimodal screen feedback. *Education and Information Technologies 26*, 4241–4260 (2021). https://doi.org/10.1007/s10639-021-10455-w
51. Vygotsky LS. *Mind in society: The development of higher psychological processes* (M. Cole, V. Jolm-Steiner, S. Scribner, & E. Souberman, Eds.). Harvard University Press; 1978 https://doi.org/10.2307/j.ctvjf9vz4

Publisher's note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.