Kalamatika: Jurnal Pendidikan Matematika

P-ISSN 2527-5615
E-ISSN 2527-5607

Journal homepage: https://kalamatika.matematika-uhamka.com/

# Developing an AI-assisted Python-based learning medium to support students' computational thinking decomposition in transformation geometry

Citra Megiana Pertiwi*, Nelly Fitriani, and Atika Ayu Rissa

**To cite this entry:**

Pertiwi, C. M., Fitriani, N., & Rissa, A. A. (2026). Developing an AI-assisted Python-based learning medium to support students' computational thinking decomposition in transformation geometry. *Kalamatika: Jurnal Pendidikan Matematika*, *11*(2), 333-366. https://doi.org/10.22236/KALAMATIKA.vol11no2.2026pp333-366

Link to the article online:
https://kalamatika.matematika-uhamka.com/index.php/kmk/article/view/888

Submitted: *Jun 11, 2026* | Revised: *Aug 26, 2026* | Accepted: *Aug 28, 2026*
Published online: *November 22, 2026*

© 2026 The Author(s). Published by FKIP Universitas Muhammadiyah Prof. Dr. HAMKA in collaboration with Indonesian Mathematics Educators (IMES).

This work is an Open Access article licensed under a Creative Commons Attribution - ShareAlike 4.0 International License.

OPEN ACCESS

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

# Developing an AI-assisted Python-based learning medium to support students' computational thinking decomposition in transformation geometry

**Citra Megiana Pertiwi[1], Nelly Fitriani[2], and Atika Ayu Rissa[3]**

[1,2]*Mathematics Education, IKIP Siliwangi, Cimahi, Indonesia*
[3]*Early Childhood Education, Pai Chai University, South Korea*

## ABSTRACT

Computational thinking (CT) is increasingly important in mathematics learning. However, empirical evidence describing students' decomposition processes in transformation geometry remains limited. This study aimed to develop and validate an AI-assisted, Python-based learning medium for transformation geometry and to describe students' CT decomposition performance and perceptions during its classroom implementation. The medium was developed using the 4D model, which consists of four stages: define, design, develop, and disseminate. It was subsequently implemented in a descriptive quantitative study involving 25 Grade 9 students from a junior high school in West Bandung Regency, Indonesia. Data were collected using expert validation sheets, four open-ended CT decomposition tasks, classroom observations, and a student response questionnaire. The material and media validation scores were 87.50% and 92.36%, respectively, indicating that the medium was highly feasible for classroom use. Students achieved an average CT decomposition score of 81.33%. They performed better in identifying and categorizing relevant geometric elements and transformation requirements than in iteratively evaluating and refining their decomposition strategies. Classroom observations showed that students decomposed transformation problems by identifying geometric elements, determining transformation parameters, organizing solution steps, and comparing intermediate results with visual outputs. However, reflective evaluation and strategy refinement were demonstrated less consistently. Students also reported positive perceptions of the medium's effectiveness, efficiency, and interactivity, with an overall response score of 79.55%. These findings indicate that the developed medium was feasible for classroom implementation and provided opportunities to explore CT decomposition in a non-coding learning environment. However, additional scaffolding is needed to strengthen students' iterative evaluation and reflective checking.

## KEYWORDS

Computational thinking; decomposition; Generative AI; Python

## CORRESPONDING AUTHOR

Citra Megiana Pertiwi ✉ citramegianapertiwi@ikipsiliwangi.ac.id 🖃 Mathematics Education, IKIP Siliwangi, Cimahi, Indonesia

## INTRODUCTION

Computational thinking (CT) is widely recognized as a critical competence for success in the digital era, as it enables individuals to solve problems systematically and adapt to technological advancements (Irawan et al., 2024; Tekdal, 2021). Computing concepts and practices are now integral to numerous professional fields and are considered essential skills in the global economy (Aydeniz, 2018; Rafiq et al., 2023; Yeni et al., 2024). In educational settings, CT serves as a foundational skill that prepares students to approach problems systematically and develop strong analytical abilities (Bilbao et al., 2024; Dagienė et al., 2024; Fauzi et al., 2025).

CT is typically characterized by indicators such as decomposition, abstraction, pattern recognition, algorithmic thinking, and evaluation or debugging, which are interconnected throughout the problem-solving process (Ocampo et al., 2024; Teng & Chung, 2025). Among these indicators, decomposition is particularly significant because it enables students to break down complex problems into smaller, well-defined sub-problems before determining appropriate solution strategies (Lv et al., 2023; Nordby et al., 2022). In practice, decomposition involves dividing complex problems into manageable components, allowing students to address individual sub-problems, reduce overall complexity, and facilitate incremental problem-solving (Bocconi et al., 2022; Tsai et al., 2022). This process is essential for effective problem-solving, algorithm development, and system modeling, making it relevant across educational and professional domains, from early education to software engineering (Fried et al., 2018; Tsai et al., 2022). Focusing on decomposition also provides a basis for designing CT assessments, as analyzing performance on this indicator can reveal students' specific strengths and areas for improvement (Nordby et al., 2022; Ocampo et al., 2024). The present study focuses specifically on decomposition because it represents a foundational process through which students identify relevant elements, recognize relationships among them, and organize complex mathematical problems into manageable sub-problems (Rich et al., 2019). This focus is particularly relevant to transformation geometry, where students must coordinate geometric objects, transformation rules, parameters, and sequential operations across visual and coordinate representations (Dahal et al., 2022; Fife et al., 2019).

Transformation geometry provides a relevant context for examining CT decomposition because tasks involving translation, reflection, rotation, and dilation require students to identify geometric objects, interpret transformation rules, determine appropriate parameters, and organize sequences of operations across visual, symbolic, and coordinate representations (Nordby et al., 2022). However, previous studies have reported that students often experience difficulties connecting visual changes on the Cartesian plane with symbolic or coordinate-based rules, selecting appropriate transformation parameters, and organizing multi-step transformation procedures (Mbusi & Luneta, 2023; Sahara et al., 2024; Turgut, 2022). These findings suggest that students' challenges extend beyond understanding transformation concepts and may also involve difficulties in decomposing complex tasks into manageable and logically connected steps. Several studies have reported that dynamic visualization tools can make problem components and intermediate results more visible,

potentially helping students organize solution steps, test predictions, compare original and transformed figures, and verify outcomes (Baihaki et al., 2022; Dahal et al., 2022; Gurmu et al., 2024; Ji et al., 2024). Therefore, structured transformation tasks supported by dynamic visualization may provide an appropriate context for examining students' decomposition processes, particularly when sequential or multiple transformations are involved (Nordby et al., 2022; Ocampo et al., 2024).

These reported difficulties highlight the need for learning media that make problem components, relationships, and intermediate transformation results more visible. By providing dynamic representations and immediate visual feedback, such media may help students organize sub-problems, test their reasoning, and revise incomplete or inappropriate decomposition strategies (Jablonski, 2024; Ji et al., 2024; Sun et al., 2024). Nevertheless, learning media explicitly designed to support CT decomposition in formal mathematics contexts remain limited and are often fragmented across topics, platforms, or specific activity designs (Lv et al., 2023; Nordby et al., 2022). Existing innovations frequently rely on educational games, programming activities, or unplugged approaches to promote CT (Chan et al., 2021; Su & Yang, 2023). However, applying these approaches to formal mathematics content requires representational support and feedback closely aligned with mathematical concepts and classroom problem-solving practices (Hooshyar et al., 2021; Perez Valdes et al., 2025; Pertiwi et al., 2025). Developing such interactive media can be challenging for teachers who lack sufficient programming expertise. In this context, generative AI can support media development by translating pedagogical requirements into initial code structures, generating prototypes, and facilitating iterative technical refinement (Groothuijsen et al., 2024; Munaye et al., 2025). However, AI-generated code requires human verification and refinement to ensure mathematical accuracy, functionality, and alignment with the intended learning objectives (Abdallah et al., 2025; Groothuijsen et al., 2024; Munaye et al., 2025).

In response to these needs, this study developed a Python-based mathematics learning medium to provide visualization and interactive features for transformation geometry tasks, with the aim of supporting students' CT decomposition. During development, generative AI was used to produce initial Python code prototypes, which were subsequently reviewed and refined through functional validation and pedagogical evaluation. Python was selected because of its extensive ecosystem for visualization and computation, as well as its established use in educational contexts for creating exploratory and simulation-based activities that clarify mathematical concepts through multiple representations (Bascuñana et

al., 2023).

Unlike CT approaches that require students to engage directly in programming, this study examines CT decomposition through students' interactions with system-computed mathematical representations rather than through direct coding activities (Chen et al., 2023; Li et al., 2022; Nordby et al., 2022). The main contribution of this study lies in demonstrating how a Python-based learning medium developed with generative AI support can provide opportunities for students to identify geometric elements, organize transformation procedures, and evaluate intermediate results. Furthermore, the study describes students' CT decomposition performance across the assessed decomposition indicators, using classroom observations to contextualize the quantitative findings.

Accordingly, this study aimed to develop and validate a Python-based learning medium with generative AI support for transformation geometry and to describe Grade 9 students' CT decomposition performance and perceptions during its classroom implementation. The following research questions guided the study:

RQ1. How was the Python-based learning medium developed with generative AI support and validated to support CT decomposition in transformation geometry?

RQ2. What is the profile of Grade 9 students' CT decomposition performance, both overall and across the assessed decomposition indicators, after using the learning medium?

RQ3. What are students' perceptions of the learning medium in terms of its effectiveness, efficiency, and interactivity?

## METHODS

This section describes the research design, the implementation of the four stages of the 4D development model, the participants and data sources, the research instruments, and the data analysis procedures employed in this study. The study integrated the development and validation of a learning medium with its implementation in a descriptive quantitative classroom study.

### *Research Design*

This study adopted a research and development (R&D) approach based on the 4D model proposed by Thiagarajan (1974), which comprises four stages: Define, Design, Develop, and Disseminate. The implementation of each stage is described in detail in the Research Procedure Using the 4D Model subsection. The development component addressed RQ1 by documenting the design, development, and validation of a Python-based learning

medium developed with generative AI support. The alignment among the media features, transformation geometry content, and CT decomposition indicators was examined during the Design and Develop stages. Expert validation was subsequently conducted to evaluate the material and media quality of the developed product.

The completed learning medium was subsequently implemented in a descriptive quantitative study involving 25 Grade 9 students from a junior high school in West Bandung Regency, Indonesia. This classroom implementation addressed RQ2 and RQ3. Students' CT decomposition performance was assessed using four open-ended transformation geometry tasks and analyzed using descriptive statistics. Classroom observations provided supplementary evidence for contextualizing students' performance across the decomposition indicators. Students' perceptions of the learning medium in terms of effectiveness, efficiency, and interactivity were collected through a response questionnaire and analyzed using percentages.

The Disseminate stage was operationalized as a limited classroom implementation rather than large-scale dissemination. As the study aimed to describe the development and feasibility of the learning medium, students' CT decomposition performance profiles, and their perceptions during implementation, it was not designed to examine causal effects. Accordingly, no pretest, control group, or inferential comparisons were included.

### *Research Procedure Using the 4D Model*

The research procedure followed the 4D model developed by Thiagarajan (1974), comprising the Define, Design, Develop, and Disseminate stages. In this study, the model was adapted to develop and validate a Python-based mathematics learning medium for transformation geometry. Details of the research instruments are presented in the Research Instruments subsection.

*Define Stage*. The Define stage focused on identifying the curricular, pedagogical, and technical requirements of the learning medium. The data sources at this stage consisted of curriculum documents and relevant empirical literature. These sources were reviewed to identify students' difficulties in transformation geometry, the requirements of CT decomposition, and the challenges associated with developing programming-based learning media. The analysis also established the learning objectives, content scope, required CT decomposition processes, and technical specifications of the medium. The outcomes of this analysis are reported in the Results section.

*Design Stage*. The Design stage translated the requirements identified during the

Define stage into a blueprint for the learning medium. The initial design was informed by an a priori analysis of relevant educational theories and empirical studies, particularly the CT decomposition framework proposed by Rich et al. (2019) and studies on interactive and dynamic visualization in geometry learning (Dahal et al., 2022; Marasabessy & Helsa, 2024). The design was subsequently refined through a posteriori analysis based on expert feedback and evidence obtained during the limited classroom implementation. Learning scenarios were structured to guide students in identifying relevant geometric elements, recognizing relationships among these elements, selecting appropriate transformation procedures, and evaluating intermediate results. Storyboards and flowcharts were prepared to represent the operational sequence of the medium. At this stage, the test items, observation indicators, and questionnaire dimensions were aligned with the research questions and the CT decomposition framework. The resulting pedagogical and technical specifications served as the basis for developing the Python-based application.

*Develop Stage*. The Develop stage involved constructing, testing, validating, and revising the learning medium. Generative AI was used to produce initial Python code prototypes based on the established pedagogical and technical specifications. The generated code was reviewed, tested, and refined by the researchers to ensure mathematical accuracy, technical functionality, and alignment with the learning objectives. The evaluation process involved checking the accuracy of transformation calculations, verifying whether the application functions operated according to the design specifications, and identifying technical or usability issues. When inaccuracies or functional problems were identified, the code and prompts were revised, and the application was retested until the specified requirements were met. The resulting product was an interactive desktop application that enabled users to visualize translation, reflection, rotation, and dilation through a graphical user interface. Material and media experts evaluated the product using validation sheets, and their ratings, comments, and recommendations informed subsequent revisions before classroom implementation.

*Disseminate Stage*. The Disseminate stage involved limited classroom implementation. The teacher facilitated the learning activities, while students operated the completed application through its graphical user interface. Students selected geometric objects and transformation types, entered the required parameters, observed changes in geometric representations, and compared the original and transformed figures. Neither the teacher nor the students were required to write Python code or interact directly with

generative AI during classroom instruction. Data on students' CT decomposition performance, observable classroom activities, and perceptions of the learning medium were collected during this stage.

### *Participants*

The study involved three material experts and three media experts during the Develop stage. The material experts evaluated the accuracy, relevance, and suitability of the transformation geometry content, whereas the media experts assessed the design, functionality, usability, technical quality, and instructional quality of the developed learning medium.

The classroom implementation involved 25 Grade 9 students from a junior high school in West Bandung Regency, Indonesia. They were selected through purposive sampling based on their prior knowledge of basic geometric concepts, participation in transformation geometry instruction, and familiarity with digital learning tools. The study was conducted with permission from the school authorities. Participation was voluntary, and the students provided their consent to participate. Their identities were anonymized during data analysis and reporting.

### *Research Instruments*

The instruments used in this study comprised a CT decomposition test, a student response questionnaire, a classroom observation sheet, and material and media validation sheets. The CT decomposition test, student response questionnaire, and classroom observation sheet were used during the limited classroom implementation, whereas the material and media validation sheets were administered before implementation to evaluate the developed learning medium.

The CT decomposition test and classroom observation sheet were developed by the researchers based on the decomposition framework proposed by Rich et al. (2019), whereas the student response questionnaire was adapted from Batubara et al. (2022). The wording and contextual content of the instruments were adjusted to the context of transformation geometry and the characteristics of Grade 9 students. Before classroom implementation, the CT decomposition test, student response questionnaire, and classroom observation sheet were reviewed by two experts in mathematics education. The expert review focused on the relevance of the items to the research objectives, alignment with the CT decomposition indicators, mathematical accuracy, clarity of wording, and suitability for Grade 9 students.

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

The instruments were subsequently revised based on the experts' feedback.

The CT decomposition test consisted of four open-ended tasks covering translation, reflection, rotation, and dilation. The tasks required students to identify relevant geometric elements, determine relationships among these elements, organize appropriate transformation procedures, and evaluate the suitability of their decomposition strategies when solving the problems. Students' written responses were assessed using an analytical rubric based on the operational indicators presented in Table 1.

**Table 1**. Operationalization of CT Decomposition Indicators in Transformation Geometry

| CT decomposition indicator | Operationalization in transformation geometry | Evidence expected in students' responses |
|---|---|---|
| Categorizing potential elements: identifying substantive elements | Identifying the original geometric object, its coordinates, the type of transformation, and the relevant parameters, such as a translation vector, reflection axis, rotation center and angle, or dilation center and scale factor. | Students identify and organize the relevant information needed to solve the transformation problem. |
| Categorizing potential elements: identifying relationships between elements | Determining the relationships among the original object, transformation rule, transformation parameters, and transformed object. | Students establish connections between the initial geometric information, the selected transformation rule, and the expected result. |
| Employing different strategies to execute a chosen decomposition | Dividing the problem into manageable steps, selecting an appropriate strategy, and systematically applying the transformation procedure. | Students organize the solution into ordered steps and apply an appropriate procedure to determine the transformed object. |
| Iteratively evaluating the utility of a specific decomposition | Checking intermediate results, comparing them with transformation properties or visual representations, and revising inappropriate steps when necessary. | Students verify their results, identify inconsistencies, and refine their decomposition strategies or calculations. |

*Note*. The CT decomposition indicators were adapted from Rich et al., (2019) and operationalized in the context of transformation geometry.

The student response questionnaire was adapted from Batubara et al. (2022) to examine students' perceptions of the effectiveness, efficiency, and interactivity of the learning medium. The original dimensions were retained, while the wording and contextual content of the items were modified to reflect transformation geometry, the features of the developed learning medium, and the characteristics of Grade 9 students. The adapted questionnaire was subsequently reviewed by two experts in mathematics education to assess its relevance, clarity, and suitability and was revised based on their feedback. The effectiveness dimension addressed students' perceptions of the medium's contribution to understanding transformation geometry and organizing problem-solving procedures. The efficiency dimension focused on ease of use, clarity of instructions, and the time required to complete the learning activities. The interactivity dimension concerned students' ability to manipulate geometric objects and transformation parameters, observe changes in visual representations, and receive immediate

visual feedback from the application.

The classroom observation sheet was developed to document the implementation of the learning medium and students' observable CT decomposition activities. The observation indicators included identifying relevant geometric elements, organizing transformation procedures, manipulating transformation parameters, comparing original and transformed objects, and evaluating intermediate results.

The material and media validation sheets were developed to evaluate the feasibility of the learning medium before classroom implementation (Hakiki et al., 2022). The material validation sheet covered content accuracy, content relevance, alignment with the learning objectives, and suitability of the transformation geometry material. The media validation sheet addressed visual design, functionality, usability, technical quality, and instructional quality.

### *Data Collection and Analysis*

Data collection and analysis were conducted across the four stages of the 4D model and aligned with the research questions. During the Define stage, data were obtained from curriculum documents and relevant empirical literature concerning transformation geometry, CT decomposition, students' learning difficulties, and the development of programming-based learning media. These documents and literature sources were reviewed descriptively to identify the learning objectives, content scope, CT decomposition requirements, and technical specifications of the learning medium.

During the Design stage, data consisted of development documentation, including learning scenarios, storyboards, flowcharts, and pedagogical and technical specifications. These documents were examined descriptively to assess the alignment among the learning objectives, transformation geometry content, CT decomposition indicators, research instruments, and proposed features of the learning medium.

During the Develop stage, data were collected using material and media validation sheets completed by the experts. The quantitative ratings were converted into percentage scores using the formula presented in Equation (1).

$$\text{Percentage score} = \left(\frac{\text{obtain score}}{\text{maximum possible score}}\right) x\ 100 \quad (1)$$

Percentage scores were calculated for each validation aspect and for the overall material and media components. The experts' comments and recommendations were reviewed descriptively and used to revise the learning medium before classroom implementation. The

development documentation and expert validation results were used to address RQ1.

During the Disseminate stage, the finalized learning medium was implemented with 25 Grade 9 students. Classroom observations were conducted while students explored translation, reflection, rotation, and dilation using the application. Following the learning activities, students completed four open-ended CT decomposition tasks and a student response questionnaire.

Students' written responses were assessed using an analytical rubric based on the CT decomposition indicators proposed by Rich et al. (2019). Each student's score was divided by the maximum possible score and multiplied by 100 to obtain a percentage score. The minimum, maximum, mean, and standard deviation were calculated for each CT decomposition indicator and for overall CT decomposition performance. Classroom observation data were organized according to the CT decomposition indicators and analyzed descriptively to contextualize students' written test performance. The test and observation data were used to address RQ2.

The student response questionnaire was administered after the classroom implementation. Students' responses were summed for each dimension and converted into percentages of the maximum possible scores. Mean percentages were calculated for the effectiveness, efficiency, and interactivity dimensions, as well as for the overall student response. The questionnaire data were used to address RQ3.

The percentage scores for expert validation, CT decomposition performance, and student responses were interpreted using the operational criteria presented in Table 2.

**Table 2**. Operational Criteria for Interpreting Percentage Scores

| Data type | Percentage range | Interpretation |
|---|---|---|
| Material and media validation | 81.00–100.00 | Very feasible |
| | 61.00–80.99 | Feasible |
| | 41.00–60.99 | Moderately feasible |
| | 21.00–40.99 | Less feasible |
| | 0.00–20.99 | Not feasible |
| CT decomposition performance | 80.00–100.00 | High |
| | 60.00–79.99 | Moderate |
| | 40.00–59.99 | Low |
| | 0.00–39.99 | Very low |
| Student responses | 81.00–100.00 | Very positive |
| | 61.00–80.99 | Positive |
| | 41.00–60.99 | Moderately positive |
| | 21.00–40.99 | Less positive |
| | 0.00–20.99 | Not positive |

Note. The percentage ranges were established as operational criteria for descriptive interpretation in this study.

### *Ethical Considerations*

The study was conducted with formal permission from the school authorities, including authorization to use anonymized student data for research and publication purposes. To ensure confidentiality, the identities of the school and all students were removed before data analysis and reporting. Only anonymized and aggregated results are presented in this article. No student names, photographs, direct quotations, or identifiable student work are included.

## RESULT AND DISCUSSION

The findings are presented according to the research questions and the stages of the 4D development model. The results addressing RQ1 include the learning requirements identified during the Define stage, the operational design of the learning medium during the Design stage, the generative AI-supported development process, the finalized transformation modules, and the expert validation results obtained during the Develop stage. The findings from the limited classroom implementation during the Disseminate stage comprise students' CT decomposition performance, addressing RQ2, and their perceptions of the learning medium in terms of effectiveness, efficiency, and interactivity, addressing RQ3.

### *Result*

#### *Development and Validation of the Python-Based Learning Medium*

The development and validation findings are presented according to the Define, Design, and Develop stages of the 4D model. These stages involved identifying the learning requirements, formulating the operational design, developing the Python-based learning medium with generative AI support, and evaluating and refining the product through expert validation.

1. Define Stage: Identification of Learning Requirements

The Define stage involved identifying the curricular, pedagogical, and technical requirements for the learning medium. The curriculum analysis indicated that Grade 9 transformation geometry covers translation, reflection, rotation, and dilation, requiring students to relate geometric objects, coordinates, transformation rules, and transformation parameters. A review of relevant empirical literature suggested that students may experience difficulties coordinating visual representations with symbolic and coordinate-based procedures and decomposing transformation problems into smaller, interconnected steps (Batubara et al., 2022; Dahal et al., 2022; Marasabessy & Helsa, 2024).

Based on this analysis, the learning medium was designed to provide dynamic visualizations of original and transformed geometric objects on a Cartesian coordinate plane. It was also expected to guide students in identifying relevant geometric elements, recognizing relationships among these elements, organizing transformation procedures into manageable steps, and evaluating intermediate results and final outcomes. In addition, the medium needed to be accessible through a graphical user interface (GUI), allowing students to explore transformation geometry without writing Python code or interacting directly with generative AI. These requirements were subsequently translated into the operational design and technical specifications during the Design stage.

2. Design Stage: Operational Design of the Learning Medium

The Design stage translated the curricular, pedagogical, and technical requirements identified during the Define stage into an operational structure for the learning medium. The operational flow was developed by the researchers rather than adopted from an existing model. Its structure was informed by the CT decomposition framework proposed by Rich et al. (2019) and by principles of interactive and dynamic visualization in geometry learning (Dahal et al., 2022; Marasabessy & Helsa, 2024).

The learning activities were organized to guide students through a sequence of processes: identifying relevant geometric elements, determining relationships among these elements, selecting an appropriate transformation procedure, specifying the required parameters, and evaluating the resulting transformation. A storyboard and flowchart were then prepared to illustrate the sequence of student interactions with the learning medium.

Figure 1 presents the operational flow of the Python-based learning medium developed during the Design stage. Students begin by selecting an initial geometric object and entering its coordinates. They then select a transformation type—translation, rotation, reflection, or dilation—and enter the corresponding transformation parameters. The application computes and visualizes the transformation result, after which students evaluate its appropriateness. If the result is considered inappropriate, students review and revise their inputs before repeating the process. If the result is considered appropriate, students analyze the changes between the original and transformed objects and complete the learning sequence.

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

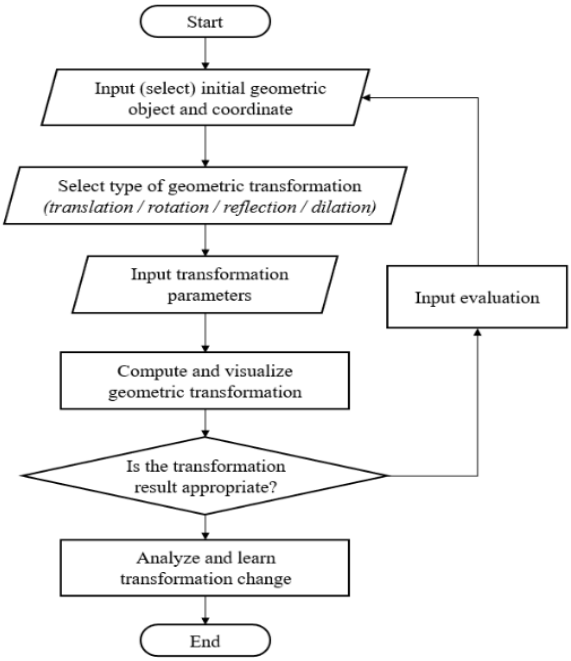



**Figure 1.** Operational Flow of the Python-Based Learning Medium

The operational flow was designed to represent CT decomposition processes in the context of transformation geometry. Selecting the geometric object, entering its coordinates, and specifying the transformation type and parameters support the identification of relevant elements and the relationships among them. Organizing these inputs into successive transformation steps represents the execution of a selected decomposition strategy, whereas evaluating the result and revising the inputs represent the iterative evaluation of the decomposition (Rich et al., 2019). The resulting storyboard and operational flow subsequently served as the basis for developing the graphical user interface (GUI) and the technical components of the learning medium during the Develop stage.

3. Developed Stage: Generative AI Supported Development Process

During the Develop stage, the pedagogical and technical specifications established during the Design stage were translated into initial Python code prototypes with the support of generative AI. The researchers formulated structured prompts describing the required mathematical functions, transformation parameters, interface components, and expected visual outputs. The generated code was not adopted without review. Instead, the researchers reviewed, tested, corrected, and refined it to ensure mathematical accuracy, technical functionality, usability, and alignment with the intended learning objectives. This process positioned generative AI as a technical development support tool rather than an autonomous developer or an instructional tool used directly by students (Memarian & Doleck, 2023; Munaye et al., 2025).

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

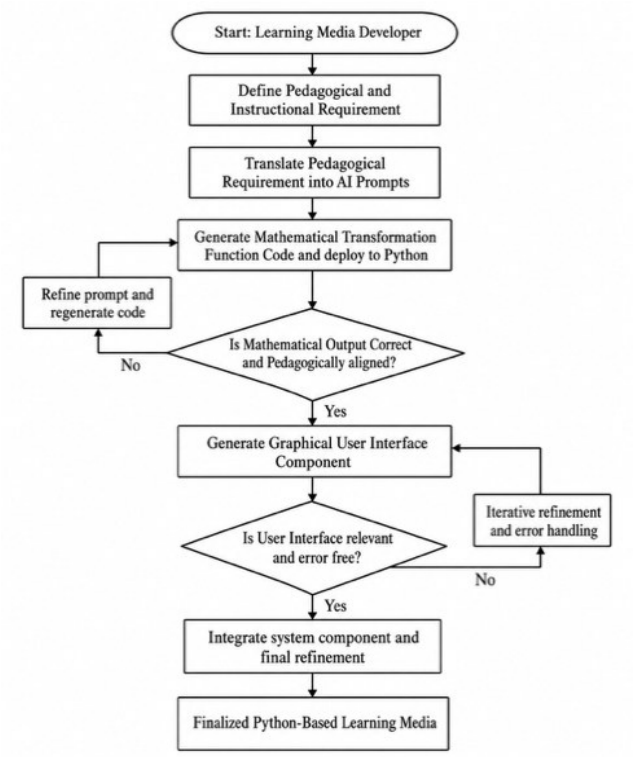



**Figure 2.** Generative AI-Supported Development Process of the Python-Based Learning Medium

Figure 2 presents the iterative prompting and refinement process used to develop the Python-based learning medium. The process began with identifying the pedagogical and instructional requirements, which were then translated into structured prompts for generative AI. These prompts were used to generate initial Python code for the mathematical transformation functions and graphical user interface components. The researchers subsequently evaluated the generated outputs in terms of mathematical correctness, pedagogical alignment, and technical functionality. When errors or inconsistencies were identified, the prompts and code were revised, and the outputs were tested again. After the transformation functions operated as intended, the graphical user interface was developed and evaluated in terms of relevance, usability, and technical reliability.

This iterative process continued until the mathematical functions, visual representations, transformation parameters, and interface components were integrated into the finalized learning medium. Thus, the development process followed a human-in-the-loop approach involving continuous evaluation, correction, and pedagogical decision-making. Generative AI supported the production of initial code prototypes, while the researchers retained responsibility for verifying, refining, and integrating the generated outputs into the completed application. The resulting product comprised four transformation modules: translation, rotation, reflection, and dilation.

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

4. Final Product and Transformation Modules

The iterative development and refinement process resulted in a Python-based learning medium comprising four transformation modules: translation, rotation, reflection, and dilation. Each module features a graphical user interface that allows students to select or enter an initial geometric object and its coordinates, specify the relevant transformation parameters, and observe the resulting transformation on a Cartesian coordinate plane. The original and transformed objects are displayed simultaneously, enabling students to directly compare their positions, shapes, and coordinate changes.

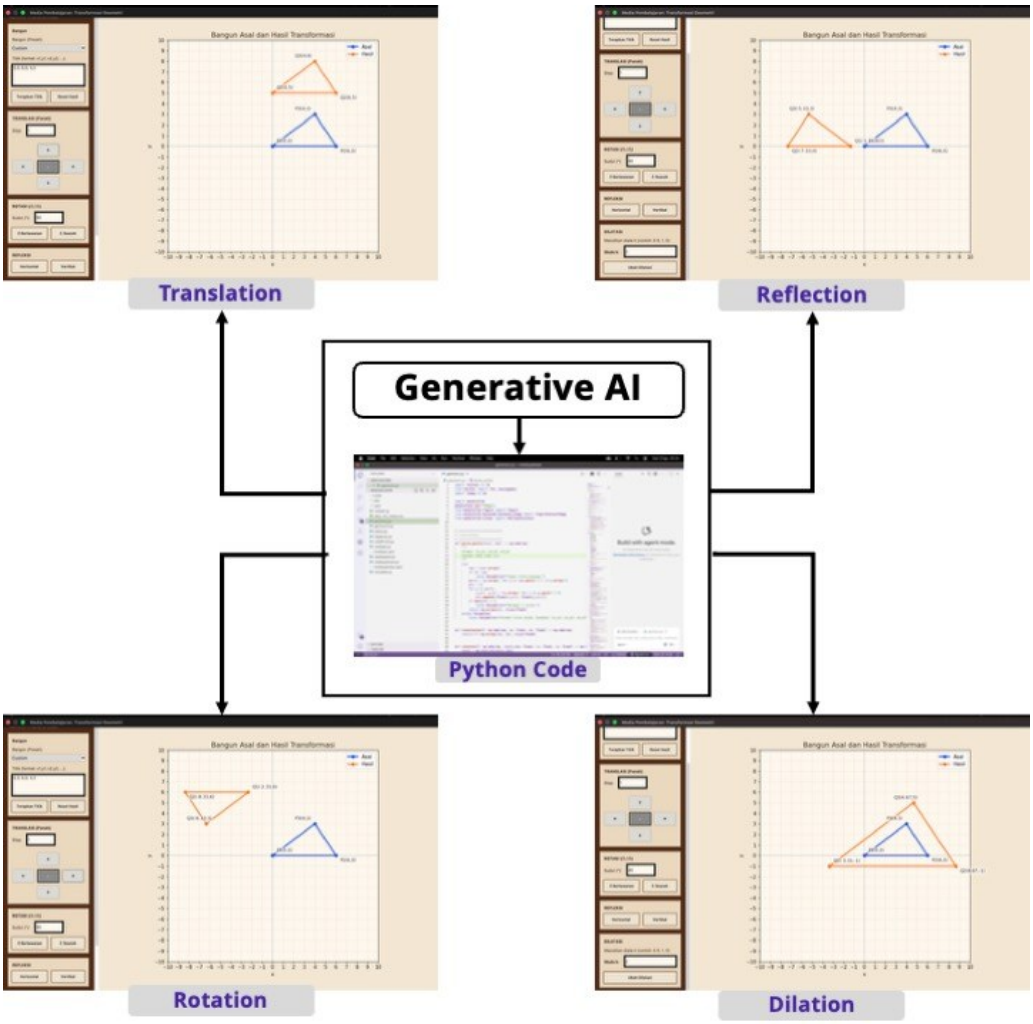



**Figure 3.** Final Transformation Modules of the Python-Based Learning Medium

Figure 3 The figure presents the four transformation modules integrated into the finalized learning medium. The translation module requires students to specify a translation vector, while the rotation module requires them to enter a center of rotation and a rotation angle. The reflection module allows students to select the relevant reflection axis, whereas the

dilation module requires them to specify a center of dilation and a scale factor. These module-specific inputs make the relevant geometric elements and relationships involved in each transformation explicit before the application computes and visualizes the results.

During classroom implementation, the teacher facilitated the learning activities and guided students in using the completed application. Students interacted with the application solely through its graphical user interface (GUI), selecting geometric objects, entering transformation parameters, observing visual outputs, and comparing the original and transformed objects. Neither the teacher nor the students were required to write Python code or interact directly with generative AI. The researchers were responsible for developing, testing, and refining the Python code before the application was implemented in the classroom.

5. Expert Validation and Product Revision

Before the limited classroom implementation, the developed learning medium was evaluated by material and media experts. Material validation focused on content accuracy and relevance, whereas media validation examined the design, functionality, and usability of the application. The validation results are presented in Table 3.

**Table 3**. Expert Validation Results for Material and Media Quality

| Component | Indicator | Percentage (%) | Interpretation |
|---|---|---|---|
| Material | Content accuracy | 86,46 | Very feasible |
| | Relevancy | 88,54 | Very feasible |
| | Overall | 87,50 | Very feasible |
| Media | Design | 94,79 | Very feasible |
| | Functionality | 90,63 | Very feasible |
| | Usability | 91,67 | Very feasible |
| | Overall | 92,36 | Very feasible |

Table 3 show that both the material and media components were categorized as very feasible. The material component obtained an overall validation score of 87.50%. Relevance received a slightly higher score (88.54%) than content accuracy (86.46%), indicating that the transformation geometry content was considered relevant to the learning objectives and appropriate for Grade 9 students.

The media component obtained an overall validation score of 92.36%. Design received the highest score (94.79%), followed by usability (91.67%) and functionality (90.63%). These findings indicate that the application's graphical interface, operational features, and technical functions were considered appropriate for classroom implementation. The higher overall media validation score also reflects the experts' positive evaluations of the application's visual presentation and ease of use.

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

In addition to providing quantitative ratings, the experts offered comments and recommendations concerning the material presentation, interface design, functionality, and usability of the learning medium. The researchers reviewed these comments and incorporated the relevant recommendations into the final product refinement before classroom implementation. The revisions were intended to improve the mathematical clarity, technical functionality, and accessibility of the application without altering its core pedagogical structure. Thus, the validation results established the feasibility of the developed product for limited classroom implementation but did not demonstrate its instructional effectiveness.

### *Students' CT Decomposition Performance during the Disseminate Stage*

During the limited classroom implementation, the finalized learning medium was used by 25 Grade 9 students in learning activities covering translation, reflection, rotation, and dilation. After completing the activities, students responded to four open-ended tasks designed to assess their CT decomposition performance. The overall and indicator-level results are presented in Table 4.

**Table 4**. Summary of Students' Computational Thinking Decomposition Performance

| Measure | *n* | Minimum (%) | Maximum (%) | Mean (%) | Standard Deviation |
|---|---|---|---|---|---|
| Categorizing potential elements | 25 | 65 | 98 | 86.20 | 7.90 |
| Employing different strategies to execute a chosen decomposition | 25 | 55 | 92 | 81,50 | 9.60 |
| Iteratively evaluating the utility of a specific decomposition | 25 | 45 | 90 | 76.29 | 10.80 |
| Overall CT decomposition | 25 | 60 | 95 | 81.33 | 8.74 |

As shown in Table 4, students achieved an overall mean CT decomposition score of 81.33%. Categorizing potential elements obtained the highest mean score (86.20%), followed by employing different strategies to execute a chosen decomposition (81.50%). Iteratively evaluating the utility of a specific decomposition obtained the lowest mean score (76.29%) and exhibited the greatest variation among students, with a standard deviation of 10.80.

Classroom observations indicated that students generally identified geometric objects, coordinates, transformation types, and relevant parameters before applying transformation procedures. Some students also compared their predictions with the visual outputs and revised their responses when the displayed results differed from their expectations. However, reflective evaluation and strategy refinement were not demonstrated consistently across students.

**Kalamatika: Jurnal Pendidikan Matematika**
Volume 11, No. 2, November 2026, pages 333-366
DOI: 10.22236/KALAMATIKA.vol11no2.2026pp333-366

### *Students' Perceptions of the Learning Medium*

Following the limited classroom implementation, students completed a response questionnaire assessing their perceptions of the developed learning medium in terms of effectiveness, efficiency, and interactivity. The results are presented in Table 5.

**Table 5**. Summary of Students' Responses to the Learning Medium

| Dimension | Mean Score (%) | Interpretation |
|---|---|---|
| Effectiveness | 80,10 | Positive |
| Efficiency | 78,70 | Positive |
| Interactivity | 79,85 | Positive |
| Overall student response | 79,55 | Positive |

As shown in Table 5, students reported positive perceptions of the developed learning medium, with an overall response score of 79.55%. Effectiveness obtained the highest mean score (80.10%), followed by interactivity (79.85%) and efficiency (78.70%). These results indicate that students generally perceived the medium as helpful for exploring transformation geometry, sufficiently interactive during the learning activities, and practical to use.

## *Discussion*

### *Development and Pedagogical Basis of the Learning Medium*

The development of the learning medium was guided primarily by pedagogical and mathematical requirements rather than by the technical capabilities of Python or generative AI. The CT decomposition framework proposed by Rich et al. (2019) was translated into a sequence of observable activities in transformation geometry. As illustrated in Figure 1, students were guided to identify the initial geometric object and its coordinates, select an appropriate transformation type, determine the required parameters, apply the transformation procedure, and evaluate the resulting representation. Identifying objects, coordinates, transformation types, and parameters reflected the categorization of substantive elements and the relationships among them. Organizing these elements into successive procedures represented the execution of a selected decomposition strategy, whereas evaluating and revising inappropriate inputs represented the iterative evaluation of the decomposition. Therefore, the operational design made decomposition visible as a sequence of connected mathematical decisions rather than leaving it as an implicit thinking process.

The graphical user interface and dynamic visualization provided representational support for these decomposition processes. The simultaneous display of the original and transformed objects allowed students to examine how changes in the translation vector, reflection axis, rotation center and angle, or dilation center and scale factor affected the

resulting figure. This design is consistent with studies showing that dynamic mathematical representations can support the exploration of geometric relationships and help learners connect visual and coordinate-based representations (Dahal et al., 2022; Gurmu et al., 2024; Ji et al., 2024; Marasabessy & Helsa, 2024). The design also reflects constructivist learning principles, according to which understanding develops through active manipulation, observation, comparison, and reflection (Do et al., 2023; Papert, 2020). However, the visualization was not intended to replace mathematical reasoning. Instead, it functioned as a source of feedback that students could use to compare their predictions with the displayed results and reconsider their procedures when inconsistencies occurred. Carefully structured didactical situations can support students' progression from tacit insights to explicit and justified mathematical understanding (Dasari et al., 2025).

The technical development process followed the pedagogical design established by the research team. The researchers first prepared visual mockups describing the intended interface and specified the mathematical and functional features required for the translation, reflection, rotation, and dilation modules. These specifications were converted into researcher-designed prompts and submitted to generative AI to produce initial Python code prototypes. As shown in Figure 2, the generated code was tested in Visual Studio Code and reviewed for mathematical accuracy, functional suitability, and conformity with the intended visual design. When the generated output did not meet the expected requirements, the researchers refined the prompts, regenerated the code, and repeated the testing process. Thus, generative AI functioned as a technical support tool within a human-in-the-loop process, while pedagogical decisions, mathematical verification, and final product refinement remained under the researchers' control. After development and validation, the finalized application was provided to the teacher for classroom use. The teacher did not need to generate or modify the Python code, while students interacted directly with the completed graphical user interface rather than with Python or generative AI.

The validation results presented in Table 3 provide evidence that the pedagogical specifications were appropriately translated into the developed product. The material component obtained an overall score of 87.50%, indicating favorable evaluations of content accuracy and relevance, while the media component obtained an overall score of 92.36%, with consistently high ratings for design, functionality, and usability. These results support the feasibility of the medium for limited classroom implementation and indicate that the experts considered its mathematical content, interface, and operational functions appropriate.

Nevertheless, the validation scores should be interpreted as evidence of product feasibility and design quality rather than as evidence that the medium caused improvements in students' CT decomposition performance.

An important insight from the development process is that the educational value of the medium did not arise merely from the use of Python or generative AI. Its pedagogical value depended on the deliberate alignment of the decomposition indicators, transformation geometry activities, interface functions, and visual feedback. The medium was designed to operate as a cycle in which students identify relevant information, organize a transformation procedure, observe the resulting representation, compare it with their expectations, and revise their decisions when necessary. This alignment provides a potentially transferable design principle for developing other CT-oriented mathematics learning media, particularly for mathematical topics in which students must coordinate multiple elements, procedures, parameters, and representations.

### *The Role of Generative AI in Media Development*

Generative AI functioned as a technical support tool during the development of the Python-based learning medium rather than as an autonomous designer or instructional agent. Before using generative AI, the research team established the pedagogical objectives, operational sequence, mathematical requirements, and intended interface of the medium. Visual mockups were prepared to represent the desired appearance and interactions of the application. The researchers then constructed prompts containing the mockups and detailed descriptions of the required features, including the input of geometric objects and coordinates, parameter controls, Cartesian plane visualization, and mathematical functions for translation, reflection, rotation, and dilation. Based on these researcher-defined specifications, generative AI produced initial Python code prototypes. This process is consistent with the potential of generative AI to assist with code generation and accelerate technical prototyping while leaving instructional decisions under human control (Memarian & Doleck, 2023; Munaye et al., 2025).

The generated code was not accepted or implemented automatically. Each prototype was executed and tested in Visual Studio Code to determine whether its interface, functions, and outputs corresponded to the intended design. The research team examined the accuracy of the transformation calculations, the consistency between numerical inputs and graphical outputs, the operation of interface components, and the occurrence of technical errors. When the code produced inaccurate mathematical results, incomplete features, inappropriate visual

displays, or operational errors, the researchers revised the prompts and requested regenerated code. This iterative cycle of prompting, testing, mathematical verification, and code refinement continued until the four transformation modules and their graphical user interfaces functioned according to the predefined specifications. Such human verification is essential because AI-generated outputs may contain errors or fail to align fully with pedagogical intentions (Abdallah et al., 2025; Groothuijsen et al., 2024; Munaye et al., 2025).

This workflow clarifies the different roles involved in the study. The research team acted as the learning-medium developer and retained responsibility for designing the mockups, constructing the prompts, evaluating the generated code, and approving the final application. Generative AI supported the conversion of these specifications into initial code and assisted with repeated code generation, but it did not determine the mathematical content, learning objectives, CT decomposition indicators, or classroom activities. After the product had been tested, refined, and validated, the finalized application was provided to the teacher for classroom use. The teacher used the completed application as an instructional medium and was not required to generate or modify the Python code during implementation. Students interacted with the graphical user interface to enter geometric information, adjust transformation parameters, and observe visual results; they did not interact directly with generative AI or Python code.

The development process suggests that generative AI can extend its role in mathematics education beyond producing instructional text by supporting the prototyping of interactive computational media. In this study, its main contribution was to accelerate the translation of predefined visual, mathematical, and functional requirements into executable Python structures and to support iterative technical refinement. Nevertheless, the findings do not establish that generative AI enables teachers with limited programming experience to develop similar media independently, because teachers were users of the finalized product rather than participants in the development process. Further research should therefore examine whether teachers can independently adopt the same mockup–prompt–test–refine workflow, what technical support they require, and whether the process can be replicated across other mathematical topics.

### *Students' CT Decomposition Performance*

The overall mean score of 81.33% indicates that students generally demonstrated CT decomposition when solving transformation geometry tasks following their use of the developed learning medium. However, the differences among the three indicators show that

decomposition was not demonstrated at the same level across all stages of problem solving. Students were more successful in identifying and organizing relevant information than in evaluating whether their initial decomposition remained useful throughout the solution process. This pattern supports the view that decomposition involves several related but distinct processes, including identifying problem elements, executing a selected decomposition strategy, and iteratively evaluating its utility (Rich et al., 2019).

Categorizing potential elements obtained the highest mean score of 86.20%. Classroom observations supported this result, as students generally began by identifying the initial geometric object, its coordinates, the transformation type, and the relevant parameters. For example, rotation tasks required students to distinguish the original coordinates, center of rotation, and rotation angle before determining the transformed figure. The learning medium made these elements visible through separate input fields and simultaneous representations of the original and transformed objects. This design may have helped students externalize the components of the problem and recognize the relationships among them. Dynamic representations have similarly been reported to support learners in connecting geometric objects, parameters, and visual outcomes across multiple representations (Dahal et al., 2022; Gurmu et al., 2024; Ji et al., 2024).

The mean score of 81.50% for employing different strategies to execute a chosen decomposition indicates that students were also generally able to organize the identified elements into successive transformation procedures. Students could determine the required inputs, apply the selected transformation rule, and follow a sequence leading to the resulting figure. Nevertheless, this score was lower than that for categorizing potential elements, suggesting that recognizing relevant information was less demanding than coordinating that information into an appropriate solution procedure. This result also needs to be interpreted carefully: it indicates students' ability to execute a selected decomposition strategy, but it does not necessarily demonstrate that every student flexibly compared or switched among several alternative strategies. Developing such flexibility may require tasks that explicitly ask students to solve the same transformation problem using different procedures and justify which decomposition is more appropriate.

Iteratively evaluating the utility of a specific decomposition obtained the lowest mean score of 76.29% and the largest standard deviation (10.80). Observation records indicated that some students compared their predictions with the visual outputs and revised coordinates or transformation parameters when inconsistencies appeared. However, this reflective checking

was not demonstrated consistently. Some students appeared to treat the displayed output as the end of the procedure rather than as evidence to be evaluated against their initial reasoning. This finding suggests that immediate visual feedback alone does not automatically lead to reflective evaluation. More advanced decomposition requires students not only to detect an incorrect result but also to examine whether the selected problem components, relationships, and procedural sequence remain appropriate. Previous research similarly indicates that evaluation, debugging, and iterative refinement generally require explicit instructional scaffolding (Nordby et al., 2022; Sun et al., 2024).

Taken together, the findings reveal a pattern in students' CT decomposition performance: identifying and categorizing problem elements was the strongest process, executing a selected decomposition strategy was somewhat less consistent, and iterative evaluation was the most challenging. The developed medium therefore appeared to provide useful representational support for making problem components and procedural steps visible, but it offered less direct support for evaluating the quality of students' decomposition decisions. Future refinements should incorporate prediction prompts, comparisons of alternative strategies, explanation fields, and reflective questions that require students to justify why a decomposition is appropriate and how it should be revised when the outcome differs from their expectations.

### *Students' Perceptions of the Learning Medium*

The student response results provide additional evidence regarding the acceptability of the developed learning medium during classroom implementation. As shown in Table 5, the overall response score was 79.55%, with all three dimensions categorized as positive. Effectiveness obtained the highest score (80.10%), followed by interactivity (79.85%) and efficiency (78.70%). These results indicate that students generally perceived the medium positively as a tool for learning transformation geometry. However, the questionnaire results represent students' perceptions and should not be interpreted as direct evidence that the medium caused improvements in their CT decomposition performance. This distinction is important because the study used a descriptive design without a comparison group or pretest–posttest measurement.

The comparatively high effectiveness score suggests that students perceived the medium as useful for supporting their exploration of transformation concepts. The application presented the original and transformed objects simultaneously and enabled students to examine the results of translation, reflection, rotation, and dilation on the Cartesian plane.

These features may have helped students connect numerical parameters with visible geometric changes. This interpretation is consistent with previous studies reporting that dynamic mathematical visualization can support students' understanding of geometric relationships and multiple representations (Dahal et al., 2022; Gurmu et al., 2024; Ji et al., 2024). Nevertheless, the positive effectiveness rating reflects perceived usefulness rather than independently measured learning effectiveness.

The interactivity score of 79.85% indicates that students responded positively to the opportunity to interact with the graphical user interface. During the learning activities, students selected geometric objects, entered coordinates, chose transformation types, adjusted the required parameters, and observed the corresponding visual results. This interaction positioned students as active users of mathematical representations rather than passive observers of completed demonstrations. Such features are consistent with constructivist learning principles, which emphasize active manipulation, observation, and meaning-making in the development of mathematical understanding (Do et al., 2023; Papert, 2020). Importantly, this interactivity occurred through the finalized application; students were not required to interact with generative AI or write Python code.

Efficiency received the lowest score among the three dimensions, although it remained within the positive category. The relatively small differences among the three scores indicate that students' perceptions were generally consistent across effectiveness, efficiency, and interactivity. However, the lower efficiency score may suggest that some aspects of operating the application required additional time, guidance, or familiarity. Because only aggregate questionnaire scores were collected, the specific reasons for this comparatively lower rating cannot be determined conclusively. Future refinements could therefore focus on simplifying coordinate and parameter entry, improving navigation instructions, and reducing unnecessary operational steps while maintaining the mathematical decision-making required for decomposition.

Taken together, the perception results suggest that the developed medium was generally accepted by students as useful, interactive, and reasonably practical for classroom exploration. The findings also show that positive perceptions were reported in a non-coding learning environment in which students could focus on geometric reasoning rather than programming syntax. Nevertheless, positive responses should be treated as complementary evidence of usability and classroom acceptance. They do not replace performance data or establish causal effectiveness.

*Educational Implications, Limitations, and Future Refinement*

The findings offer several implications for the design of CT-oriented mathematics learning media. First, computational thinking can be supported through interaction with dynamic mathematical representations without requiring students to write programming code. In this study, students engaged in decomposition by identifying geometric elements, selecting transformation procedures, entering relevant parameters, and examining visual results through the graphical user interface. This suggests that the pedagogical value of the medium lies in making mathematical components, relationships, and intermediate outcomes visible. Similar design principles may be applied to other mathematical topics that require students to coordinate multiple representations and procedural steps, such as functions, vectors, analytic geometry, statistics, and probability simulations. However, these broader applications remain potential extensions and were not examined directly in the present study.

The findings also indicate that visual feedback should not be considered sufficient support for all aspects of decomposition. Although the medium made geometric elements and transformation results visible, students' comparatively lower performance in iterative evaluation suggests that displaying an output does not automatically encourage them to evaluate the suitability of their initial strategy. Future versions should therefore incorporate more explicit reflective scaffolding, such as prediction prompts before visualization, comparisons of alternative procedures, explanation fields, and questions requiring students to justify their revisions. A history of parameter changes or a feature that allows students to compare several transformation attempts may also help make the process of evaluation and refinement more explicit.

Several limitations should be considered when interpreting the findings. The study involved only 25 Grade 9 students in a limited classroom implementation, which restricts the generalizability of the results. The descriptive design did not include a control group, pretest–posttest comparison, or inferential analysis. Therefore, the findings describe students' CT decomposition profiles and perceptions but do not establish that the learning medium caused improvements. Future studies should employ larger and more diverse samples and use experimental or quasi-experimental designs with appropriate comparison groups and pretest–posttest measures to examine the medium's effectiveness more rigorously. In addition, the questionnaire reported aggregate perceptions of effectiveness, efficiency, and interactivity, so the specific reasons behind students' ratings, particularly the comparatively lower efficiency

score, could not be determined. The application was also developed for desktop or laptop use, which may limit its implementation in contexts with restricted technological infrastructure.

A further limitation concerns the role of teachers in media development. Although generative AI assisted the research team in producing and refining Python code, the study did not examine whether teachers with limited programming experience could independently use the same mockup–prompt–test–refine workflow. The teacher used the finalized product rather than developing or modifying its code. Accordingly, claims that generative AI reduces programming barriers for teachers should be regarded as potential implications rather than empirical findings. Future research should involve teachers directly as media developers, examine the usability of the workflow, and identify the technical and pedagogical support required for independent adoption.

Based on these limitations, future research and refinement may proceed in three directions. The first is pedagogical refinement through stronger scaffolding for iterative evaluation and strategy comparison. The second is technical refinement through the development of web-based or cross-platform versions that can be accessed on a wider range of devices. The third is methodological refinement through larger and more diverse samples, teacher-development trials, and experimental or quasi-experimental designs incorporating appropriate comparison groups and pretest–posttest measures. These directions would provide stronger evidence regarding the effectiveness, scalability, and transferability of AI-assisted Python-based learning media across mathematical topics and educational settings.

## CONCLUSION

This study developed and validated an AI-assisted, Python-based learning medium for transformation geometry using the 4D model. During development, generative AI was used to generate initial Python code prototypes based on researcher-defined pedagogical and technical specifications, while the researchers retained responsibility for testing, mathematical verification, refinement, and final integration of the application. Expert validation indicated that the material and media components were very feasible, with overall scores of 87.50% and 92.36%, respectively. During the limited classroom implementation, Grade 9 students achieved an overall mean CT decomposition score of 81.33%. Students performed best in categorizing relevant geometric elements, followed by employing a selected decomposition strategy, while iteratively evaluating and refining the utility of a decomposition was the most challenging process. Classroom observations showed a similar pattern: students generally identified geometric elements, transformation types, and parameters, whereas reflective

checking and strategy revision were demonstrated less consistently. Students also reported positive perceptions of the medium's effectiveness, efficiency, and interactivity, with an overall response score of 79.55%. These findings indicate that the developed medium was feasible for classroom implementation and provided a non-coding environment for exploring CT decomposition through dynamic mathematical representations.

The study contributes to CT-oriented mathematics education by illustrating how dynamic visualization and a human-in-the-loop, AI-assisted development process can be aligned with decomposition activities without requiring students to write Python code or interact directly with generative AI. However, the descriptive design, limited sample, aggregate questionnaire data, desktop-based implementation, and absence of direct examination of teachers as media developers constrain the interpretation and broader application of the findings. Future research should incorporate stronger scaffolding for prediction, comparison, justification, and iterative evaluation; involve larger and more diverse samples; employ experimental or quasi-experimental designs with appropriate comparison groups and pretest–posttest measures; develop web-based or cross-platform versions of the learning medium; and investigate whether teachers can independently apply the mockup–prompt–test–refine workflow to develop similar mathematics learning media.

## ACKNOWLEDGMENTS

The authors would like to express their sincere gratitude to all those who contributed to this study through their valuable input, participation, support, and cooperation during the development and implementation of the learning medium.

## REFERENCES

Abdallah, N., Katmah, R., Khalaf, K., & Jelinek, H. F. (2025). Systematic review of ChatGPT in higher education: Navigating impact on learning, wellbeing, and collaboration. *Social Sciences & Humanities Open*, *12*, Article 101866. https://doi.org/10.1016/j.ssaho.2025.101866

Aydeniz, M. (2018). Integrating computational thinking in school curriculum. In *Computational thinking in the STEM disciplines: Foundations and research highlights* (pp. 253–277). Springer. https://doi.org/10.1007/978-3-319-93566-9_13

Baihaki, B., Djamilah, S., & Lazwardi, A. (2022). Developing interactive learning media based on Adobe Animate Applications for geometry transformation. *Kalamatika:*

*Jurnal Pendidikan Matematika*, *7*(2), 191–206. https://doi.org/10.22236/KALAMATIKA.vol7no2.2022pp191-206

Bascuñana, J., León, S., González-Miquel, M., González, E. J., & Ramírez, J. (2023). Impact of Jupyter Notebook as a tool to enhance the learning process in chemical engineering modules. *Education for Chemical Engineers*, *44*, 155–163. https://doi.org/10.1016/j.ece.2023.06.001

Batubara, H. H., Sumantri, M. S., & Marini, A. (2022). Developing an Android-Based E-Textbook to Improve Learning Media Course Outcomes. *International Journal of Interactive Mobile Technologies*, *16*(17). https://doi.org/10.3991/ijim.v16i17.33137

Bilbao, J., Bravo, E., Garcia, O., Rebollar, C., Laakso, M.-J., Kaarto, H., Lehtonen, D., Parviainen, M., Jankauskienė, A., & Pears, A. (2024). Introducing computational thinking and algebraic thinking in the European educational systems. *International Journal of Education and Information Technologies.*, *18*, 11–19. https://doi.org/10.46300/9109.2024.18.2

Bocconi, S., Chioccariello, A., Kampylis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M., Jasutė, E., & Malagoli, C. (2022). *Reviewing computational thinking in compulsory education: State of play and practices from computing education*. https://doi.org/10.2760/126955

Chan, S.-W., Looi, C.-K., Ho, W. K., Huang, W., Seow, P., & Wu, L. (2021). Learning number patterns through computational thinking activities: A Rasch model analysis. *Heliyon*, *7*(9). https://doi.org/10.1016/j.heliyon.2021.e07922

Chen, P., Yang, D., Metwally, A. H. S., Lavonen, J., & Wang, X. (2023). Fostering computational thinking through unplugged activities: A systematic literature review and meta-analysis. *International Journal of STEM Education*, *10*(1), 47. https://doi.org/10.1186/s40594-023-00434-7

Dagienė, V., Kampylis, P., Giannoutsou, N., Engelhardt, K., Malagoli, C., & Bocconi, S. (2024). Fostering computational thinking in compulsory education in Europe: a

multiple case study. *Baltic Journal of Modern Computing.*, *12*(2), 189–221. https://doi.org/10.22364/bjmc.2024.12.2.05

Dahal, N., Pant, B. P., Shrestha, I. M., & Manandhar, N. K. (2022). Use of GeoGebra in teaching and learning geometric transformation in school mathematics. *Int. J. Interact. Mob. Technol.*, *16*(8), 65–78. https://doi.org/10.3991/ijim.v16i08.29575

Dasari, D., Hendriyanto, A., Sahara, S., Ariawan, I. P. W., Muhaimin, L. H., & Pahmi, S. (2025). Crafting epistemic knowledge in statistics education via didactical engineering integrating the theory of didactical situations and the SECI model. *Discover Sustainability*, *6*(1), 928. https://doi.org/10.1007/s43621-025-01838-x

Do, H.-N., Do, B. N., & Nguyen, M. H. (2023). 3How do constructivism learning environments generate better motivation and learning strategies? The Design Science Approach. *Heliyon*, *9*(12). https://doi.org/10.1016/j.heliyon.2023.e22862

Fauzi, A. L., Kusumah, Y. S., Nurlaelah, E., & Juandi, D. (2025). Computational thinking education in K–12 artificial intelligence literacy and physical computing [Review of the book *Computational thinking education in K–12: Artificial intelligence literacy and physical computing*, edited by S.-C. Kong & H. Abelson]. *Education 3–13, 53*(4), 687–689. https://doi.org/10.1080/03004279.2023.2271019

Fife, J. H., James, K., & Bauer, M. (2019). A learning progression for geometric transformations. *ETS Research Report Series*, *2019*(1), 1–16. https://doi.org/10.1002/ets2.12236

Fried, D., Legay, A., Ouaknine, J., & Vardi, M. Y. (2018). Sequential relational decomposition. *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*, 432–441. https://doi.org/10.1145/3209108.3209203

Groothuijsen, S., van den Beemt, A., Remmers, J. C., & van Meeuwen, L. W. (2024). AI chatbots in programming education: Students' use in a scientific computing course and consequences for learning. *Computers and Education: Artificial Intelligence*, *7*, Article 100290. https://doi.org/10.1016/j.caeai.2024.100290

Gurmu, F., Tuge, C., & Hunde, A. B. (2024). Effects of GeoGebra-assisted instructional methods on students' conceptual understanding of geometry. *Cogent Education*, *11*(1), Article 2379745. https://doi.org/10.1080/2331186X.2024.2379745

Hakiki, R., Muchson, M., Sulistina, O., & Febriana, A. (2022). The Development of Learning Media Based on Augmented Reality, Hologram, and Ludo Game on The Topic of Molecular Shapes. *Int. J. Interact. Mob. Technol.*, *16*(4), 70–84. https://doi.org/10.3991/ijim.v16i04.28989

Hooshyar, D., Malva, L., Yang, Y., Pedaste, M., Wang, M., & Lim, H. (2021). An adaptive educational computer game: Effects on students' knowledge and learning attitude in computational thinking. *Computers in Human Behavior*, *114*, Article 106575. https://doi.org/10.1016/j.chb.2020.106575

Irawan, E., Rosjanuardi, R., & Prabawanto, S. (2024). Research trends of computational thinking in mathematics learning: A bibliometric analysis from 2009 to 2023. *Eurasia Journal of Mathematics, Science and Technology Education*, *20*(3), Article em2417. https://doi.org/10.29333/ejmste/14343

Jablonski, S. (2024). Students' verification and elaboration in outdoor mathematics: The role of digital feedback in MathCityMap. *Digital Experiences in Mathematics Education*, *10*(1), 132–157. https://doi.org/10.1007/s40751-024-00137-w

Ji, Z., Guo, K., & Song, S. (2024). Effects of dynamic mathematical software on students' performance: A three-level meta-analysis. *Journal of Educational Computing Research*, *62*(4), 815–840. https://doi.org/10.1177/07356331241226594

Li, F., Wang, X., He, X., Cheng, L., & Wang, Y. (2022). The effectiveness of unplugged activities and programming exercises in computational thinking education: A Meta-analysis. *Education and Information Technologies*, *27*(6), 7993–8013. https://doi.org/10.1007/s10639-022-10915-x

Lv, L., Zhong, B., & Liu, X. (2023). A literature review on the empirical studies of the integration of mathematics and computational thinking. *Education and Information Technologies*, *28*(7), 8171–8193. https://doi.org/10.1007/s10639-022-11518-2

Marasabessy, R., & Helsa, Y. (2024). Fostering spatial visualization in GeoGebra-assisted geometry lesson: A systematic review and meta-analysis. *Eurasia Journal of Mathematics, Science and Technology Education*, *20*(9), Article em2509. https://doi.org/10.29333/ejmste/15170

Mbusi, N., & Luneta, K. (2023). Implementation of an intervention program to enhance student teachers' active learning in transformation geometry. *SAGE Open*, *13*(2), Article 21582440231179440. https://doi.org/10.1177/21582440231179440

Memarian, B., & Doleck, T. (2023). ChatGPT in education: Methods, potentials, and limitations. *Computers in Human Behavior: Artificial Humans*, *1*(2), Article 100022. https://doi.org/10.1016/j.chbah.2023.100022

Munaye, Y. Y., Admass, W., Belayneh, Y., Molla, A., & Asmare, M. (2025). ChatGPT in Education: A Systematic Review on Opportunities, Challenges, and Future Directions. *Algorithms*, *18*(6), 352. https://doi.org/10.3390/a18060352

Nordby, S. K., Bjerke, A. H., & Mifsud, L. (2022). Computational thinking in the primary mathematics classroom: A systematic review. *Digital Experiences in Mathematics Education*, *8*(1), 27–49. https://doi.org/10.1007/s40751-022-00102-5

Ocampo, L. M., Corrales-Álvarez, M., Cardona-Torres, S. A., & Zapata-Cáceres, M. (2024). Systematic review of instruments to assess computational thinking in early years of schooling. *Education Sciences*, *14*(10), 1124. https://doi.org/10.3390/educsci14101124

Papert, S. A. (2020). *Mindstorms: Children, computers, and powerful ideas*. Basic books.

Perez Valdes, G. P., Boude Figueredo, O., & Vargas Sanchez, A. D. (2025). Integrating computational thinking in children aged 3 to 6: challenges and opportunities in early childhood education. *Frontiers in Education*, *10*, Article 1535135. https://doi.org/10.3389/feduc.2025.1535135

Pertiwi, C. M., Rohaendi, N., & Rahayu, G. D. S. (2025). Problem solving approach and learning media based on vba for excel as an effort to improve the mathematical

communication skills of elementary school students. *Jurnal Cakrawala Pendas*, *11*(2), 418–430. https://doi.org/10.31949/jcp.v11i2.13237

Rafiq, A. A., Triyono, M. B., Djatmiko, I. W., Wardani, R., & Köhler, T. (2023). Mapping the evolution of computational thinking in education: A bibliometrics analysis of scopus database from 1987 to 2023. *Informatics in Education*, *22*(4), 691–724. https://doi.org/10.15388/infedu.2023.29

Rich, P. J., Egan, G., & Ellsworth, J. (2019). A framework for decomposition in computational thinking. *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, 416–421. https://doi.org/10.1145/3304221.3319793

Sahara, S., Dolk, M., Hendriyanto, A., Kusmayadi, T. A., & Fitriana, L. (2024). Transformation Geometry in Eleventh Grade Using Digital Manipulative Batik Activities. *Journal on Mathematics Education*, *15*(1), 55–78. https://doi.org/10.22342/jme.v15i1.pp55-78

Su, J., & Yang, W. (2023). A systematic review of integrating computational thinking in early childhood education. *Computers and Education Open*, *4*, Article 100122. https://doi.org/10.1016/j.caeo.2023.100122

Sun, C., Yang, S., & Becker, B. (2024). Debugging in computational thinking: A meta-analysis on the effects of interventions on debugging skills. *Journal of Educational Computing Research*, *62*(4), 867–901. https://doi.org/10.1177/07356331241227793

Tekdal, M. (2021). Trends and development in research on computational thinking. *Education and Information Technologies*, *26*(5), 6499–6529. https://doi.org/10.1007/s10639-021-10617-w

Teng, K., & Chung, G. K. W. K. (2025). Measuring Children's Computational Thinking and Problem-Solving in a Block-Based Programming Game. *Education Sciences*, *15*(1), 51. https://doi.org/10.3390/educsci15010051

Thiagarajan, S. (1974). *Instructional development for training teachers of exceptional children: A sourcebook.*

Tsai, M.-J., Liang, J.-C., Lee, S. W.-Y., & Hsu, C.-Y. (2022). Structural validation for the developmental model of computational thinking. *Journal of Educational Computing Research*, *60*(1), 56–73. https://doi.org/10.1177/07356331211017794

Turgut, M. (2022). Reinventing Geometric Linear Transformations in a Dynamic Geometry Environment: Multimodal Analysis of Student Reasoning: M. Turgut. *International Journal of Science and Mathematics Education*, *20*(6), 1203–1223. https://doi.org/10.1007/s10763-021-10185-y

Yeni, S., Nijenhuis-Voogt, J., Saeli, M., Barendsen, E., & Hermans, F. (2024). Computational thinking integrated in school subjects–A cross-case analysis of students' experiences. *International Journal of Child-Computer Interaction*, *42*, Article 100696. https://doi.org/10.1016/j.ijcci.2024.100696