

Design and Development of Matter Module with the Integration of Computational Thinking by Employing ADDIE Model

Noraini Lapawi

Institut Pendidikan Guru Kampus Pulau Pinang

Hazrati Husnin (Corresponding Author)

Universiti Kebangsaan Malaysia,

Email: hazrati@ukm.edu.my

To Link this Article: <http://dx.doi.org/10.6007/IJARBSS/v11-i12/11757> DOI:10.6007/IJARBSS/v11-i12/11757

Published Date: 10 December 2021

Abstract

This study aims to explore the design and development process of the Matter module for secondary school students. The needs analysis shows a similar perception between students and teachers who described the matter topic as complex due to the abstract nature of the Matter topic. This module emphasises the use of animation in visualising particles in different matter states. The module was designed to make learning more active and meaningful by integrating computational thinking skills through cooperative and project-based learning. Through this approach, students are actively involved in building animation projects to visualise the movement of particles by applying computational thinking concepts (abstraction, decomposition, generalisation, algorithm, and evaluation) as they engage with programming in Scratch. Thus, to systematically design and develop the Matter module, the ADDIE model was employed. The design and development of the module comprise five phases, Analysis, Design, Development, Implementation, and Evaluation. The module was evaluated by experts using a formative evaluation approach. The results show that the validity of the Matter module is high and can support the integration of computational thinking skills.

Keywords: Computational thinking, ADDIE Model, Module Development, Science, Scratch, Project-Based Learning, Cooperative Learning

Introduction

Computational thinking (CT) skills have been introduced into the curriculum system in line with the efforts to teach problem-solving skills at the early stages of schooling. Education at the primary, secondary, and tertiary levels is an essential phase in the formation of students' personalities as well as cognitive, affective, and psychomotor capabilities. Computational thinking involves high-level and systematic thinking processes in analysing problems and

ultimately leads to practical problem solving based on computer programming concepts such as decomposition, abstraction, and algorithmic thinking. Computational thinking can be applied in teaching and learning (Wing, 2006) and has been proven effective in helping to improve students' logical thinking and problem-solving. They will have a more significant impact on student's academic mastery, especially in problem-solving skills.

This study uses the Matter topic for secondary school as content for the design and development of the module. Based on the feasibility study conducted earlier, the Matter topic was one of the most complex topics in the science syllabus. Among the causes of the problems identified is teaching the Matter topic covering abstract concepts. The concepts include the state of matter, the characteristics of the three states of matter, and the change of state of matter. The difficulty of visualising these abstract concepts often invites problems to teachers and students and, in turn, often creates misconceptions of the matter fundamental concepts. The study of Md. Dalhar and Ali (2009) explained that it is challenging for most students to see the matter elements through a rough view and imagination and difficult to describe. Based on the literature review, researchers have suggested that the teaching and learning approach through computer animation for Matter topics is a way to address difficulties in learners even this helps students master the subjects more effectively (Abdul Wahab, 2006). The study of Wan Muda (2017) supports the effectiveness of visualisation methods in teaching and learning, such as visual learning, in enhancing students' understanding more effectively and meaningfully. However, learning through animation might be too passive as the learners only process the information presented to them. Thus, this module requires learners to create animation using Scratch with computational thinking integration to promote active learning.

The computational thinking skills were currently embedded in the syllabus since there is no specific subject on it. However, the teaching and learning of topics in schools, especially Mathematics and Science, is still questioned regarding how this learning experience can foster computational thinking skills. Efforts to build these skills are still limited due to teachers' lack of exposure to computational skills (Senin & Nasri, 2019), particularly how this skill can be integrated into the learning content and activities (Rahayu & Osman, 2019). Furthermore, most studies emphasise using computational thinking as an intervention (Khasyyatillah & Osman, 2019; Chongo et al., 2021) rather than discussing the design aspect of developing a module that integrates computational thinking in-depth. Therefore, this study aimed to explore the design and development of a Matter module with the integration of computational thinking by employing the ADDIE instructional design model.

The Design and Development of Matter Module

The procedure that underlies the design and development of the Matter module is the ADDIE Model. The ADDIE model is a generic model that can develop various instructional products (Shariffudin, 2007). ADDIE model was chosen to support this module's development as its phases are relevant and essential to the development process. ADDIE model follows an essential step-by-step process in instructional design and development, and at the same time, allows each phase to be evaluated for improvement purposes. This model has been documented as a good model (Wilson et al., 1993). There were five phases involved in the design and development based on the ADDIE model:

Analysis Phase

During this phase, the analysis was conducted of three stages: analysis on learners and teachers; needs assessment; and analysis on the elements that need to be included in the module. From the analysis, the Matter topic was the most challenging in Secondary 1 science syllabus. Most respondents also agreed that the topic should be delivered via printed Module and PowerPoint, employing collaborative learning, simulation, and visualisation. Since simulation and visualisation elements were suggested in the needs analysis, computer animation was seen as relevant in this study.

Using simulation and visualisation will help deliver the content efficiently and develop learners' understanding (Muda, 2017). Therefore, a questionnaire was administered to 72 students to identify students' prerequisite skills of using technology, i.e., computer and smartphone. Since learners may be more passive when watching the animation (simulation and visualisation) solely, the questionnaire also seeks to identify students' prior knowledge on programming applications and students' experience involving programming activities such as robotic competition. Presumably, with the integration of programming applications such as Scratch and Python, the learners will be more active in learning as they create the animation using programming. The findings of the analysis are presented in the table below.

Table 1:

Total Hours (in a week) of using Computer

Total hours (in a week)	Frequency	Percentage
Less than one hour	18	25
1 to 10 hours	32	44
11 to 20 hours	9	13
21 to 30 hours	6	8
31 to 40 hours	5	7
More than 40 hours	2	3
Total	72	100

Table 2:

Total Hours (in a week) of using Smartphone (online)

Total hours (in a week)	Frequency	Percentage
Less than one hour	14	19
1 to 10 hours	11	15
11 to 20 hours	21	29
21 to 30 hours	13	18
31 to 40 hours	4	6
More than 40 hours	9	13
Total	72	100

Table 1 shows that most students (44%) spent their time using computers 1 to 10 hours a week (2 hours at least). Meanwhile, Table 2 shows that 29% of students get online via smartphones for 11 to 20 hours (at least 2 hours a day).

Table 3:

Experience in using Computer Programming Applications

Gender	Yes		No	
	Frequency	%	Frequency	%
Male	19	68	9	32
Female	26	59	18	41
Total	45	63	27	37

Table 4:

Experience of Following Robotic Competition

Jantina	Yes		No	
	Frequency	%	Frequency	%
Male	4	14	24	86
Female	1	2	43	98
Total	5	7	67	93

Table 3 shows that 63% of the students had used programming applications such as Scratch, Python, and Arduino. Table 4 indicates that 7% of the students had participated in robotics competitions before. Although the percentage of students who participate in robotics competitions is relatively low, at least the percentage using computer programming is still moderate. It could also be said that not everyone had the opportunity to participate in such a competition. Nevertheless, the result indicates that the students have prior knowledge and experience and can use computer programming (Scratch, Python, Arduino). Thus, based on the analysis, the Matter module was suggested to be carried out by integrating computational thinking via Scratch programming. Scratch has been proven to develop problem-solving skills (Lee & Khalid, 2018) since it supports an imaging-rich programming interface that allows students to create animations and games using block programming.

Design Phase

At this stage, the design aspect of the module was planned. The learning objectives, learning activities, and assessments to be carried out during the learning process using this module were aligned to the curriculum standards. This module involves students working in a group to produce projects related to essential concepts in Matter topics. Therefore, cooperative learning and project-based learning approaches are used in this module. Cooperative learning substantially impacts academic achievement and helps solve problems (Liu & Yang, 2021). The approach complements the way CT skills can be integrated into teaching and learning. Therefore, to conduct this study, five computational thinking skills were selected to be integrated into this module: decomposition, generalisation, abstraction, algorithm, and evaluation (Csizmadia et al., 2015; Selby & Woollard, 2014). Meanwhile, Project-Based Learning (PBL) encourages students to think critically and creatively in problem-solving (Shin et al., 2021). The project-based learning approach was based on scaffolding; with guidance by peers and teachers in the Zone of Proximal Development (ZPD), students can solve problems that involve higher-order thinking skills (Azmi & Ummah, 2021).

Development Phase

In the development phase, the actual module was developed based on the planning made in the design phase. This includes the selection of teaching materials, teaching activities, and

appropriate media. This module integrates computational thinking skills by combining plugged-in computational thinking skills. The plugged-in module is implemented using a computer through Scratch 3.0 computer programming. The Matter module is comprised of two sub-modules, as described in Table 5.

Table 5:

Activities in Matter module - Module 1 and Module 2

Module	Descriptions of the activities
Module 1	In Module 1, the activity emphasises the three states of matter, the arrangement and movement of particles, and relates them to the physical properties of liquids, gases, and solids. Students will compare the differences of the three states of matter based on kinetic theory according to the arrangement and motion of the particles.
Module 2	In Module 2, the details of the activities are that students will describe the changes of three states of matter according to the motion of their particles caused by the release and absorption of heat by kinetic theory. Students will explain with examples what changes the three states of matter observed based on everyday life situations.

As described in the Design phase, this module employed cooperative learning and project-based learning as strategies. This module also uses the Scratch application to make the teaching and learning process more interesting while cultivating computational thinking and ICT integration in Matter topics. Project-based learning provides opportunities for students to work in groups to carry out hands-on activities, stimulating the construction of concepts among students (Bicer et al., 2015). Table 6 and Table 7 explain the learning activities in the module concerning the project and the computational thinking skills.

Table 6:

Learning Task/Activities in Module 1

Module	Learning task/activities	CT skills	Descriptions on CT integration
Module 1	Students are asked to give examples of materials, categorise three states of matter, and draw the arrangement of particles using Scratch. At the same time, students are asked to state the physical state of the matter.	Decomposition Generalisation Algorithm Evaluation	<ul style="list-style-type: none"> Students break the problem into manageable sub-tasks to determine the type of material based on the three states of matter and the arrangement of the particles. Students identify the pattern or type of material according to the categories of the three states of matter (solid, liquid, and gas) and the arrangement of its particles. Students arrange the steps for a material based on the three

-
- states of matter and draw the arrangement of its particles.
 - Students evaluate the project by observing the movement of the particles for each of the three states and providing examples of the types of materials specified.
-

Table 7:
Learning Task/Activities in Module 1

Module	Learning task/activities	CT skills	Descriptions on CT integration
Module 2	<p>Students are asked to construct and produce animations in groups to describe changes in the state of matter in daily life and examples using Scratch.</p> <p>Question: A student placed a glass of cold water on the table. After a while, the student noticed liquid droplets formed on the outer surface of the glass. Using Scratch, describe the changes in matter and compare the states of the two.</p>	<p>Decomposition</p> <p>Algorithm</p> <p>Abstraction</p> <p>Evaluation</p>	<ul style="list-style-type: none"> • Students break the problem into manageable tasks. The first step is to determine what material changes are taking place. Then compare and contrast the state of change of the two matters. • Students arrange the steps in advance, for example, the arrangement of particles and the movement of particles for the change of matter. • Based on some of the steps listed earlier in developing the Scratch project, students select only the essential steps to determine the changes in matter and compare the differences in the state of the two materials. • After the animation project is produced, students will self-evaluate the animation, i.e., the changes in the material that occur, and compare the differences between the two materials.

Throughout the Science Module development process, formative evaluation was also carried out to improve the quality of the module. All feedbacks and improvements were taken into consideration. A more detailed explanation of the evaluation phase is described in the evaluation phase.

Implementation Phase

The implementation phase considers the findings that have been obtained from the beginning of the phase, namely the first phase (analysis), the second phase (design), and the third phase (module development). According to (Seawright, 2003), among the issues taken into account during the implementation phase is the issue of implementation costs in the use of selected media, infrastructure issues in terms of hardware limitations, software limitations, bandwidth limitations, and internet access. Besides that, the implementation phase also concerned the module's duration, which primarily depends on school hours (Abd Razak et al., 2020).

Other aspects such as the schedule of school computer labs and other facilities are also considered in this study. The permission to use the computer lab was applied from the administrator before implementing the module. The school's technical assistants and ICT teachers also assisted during implementing hands-on Scratch programming workshops and other activities during the study. In this phase, science teachers involved in delivering this module are given a three-day course to receive training related to the modules developed and hands-on training in Scratch programming as a tool to be used along with the module.

Evaluation Phase

Throughout the module development process, formative assessment is carried out to improve and enhance the modules' effectiveness. At an early stage, formative evaluation was carried by appointed experts. A questionnaire to evaluate and validate the module's content was given to six experts and science teachers. The module was evaluated based on content, learning activities, CT skills, language, and terminology. Table 8 shows in detail the interpretation of the mean scores.

Table 8:

Interpretation of Mean Score

Mean Score	Interpretation
1.00 – 1.89	Very low
1.90 – 2.69	Low
2.70 – 3.49	Moderate
3.50 – 4.29	High
4.50 – 5.00	Very High

The module coefficient of validity was calculated based on the below formula. According to Sidek and Jamaludin (2005), if the value of the coefficient is equal to or greater than 0.70, then the validity of the value of the coefficient is high. The determination of the correlation coefficient was calculated using the following formula:

$$\text{Content validity coefficient} = \frac{\text{Total score}}{\text{Maximum score}}$$

The analysis of content correlation coefficients and mean item scores for the content validity of the Matter Module are shown in Table 9. The findings showed the mean for all items and correlation coefficients exceeded 4.00 and 0.70. So, this indicates that the validity of the content in the Matter Module is high.

Table 9:

Analysis of Correlation Coefficients and Mean Item Scores for the Validity of the Content of the Module

No	Item	Mean	Content correlation coefficient
Content			
1	The teaching and learning strategy prioritise a constructivist approach.	5.00	1.00
2	The teaching and learning strategy prioritise the constructionism approach.	5.00	1.00
3	The content relates the knowledge, values , and skills learned to their work and daily life application.	5.00	1.00
4	Knowledge and skills are introduced from simple to complex levels.	5.00	1.00
5	Material processing is planned and easy to understand and follow.	5.00	1.00
6	Content includes knowledge and skills were aligned with the curriculum.	5.00	1.00
7	The concepts, terms, and facts are correct and up to date	5.00	1.00
8	Concepts, terms, and facts are consistent.	4.50	0.90
Activities			
9	Activities are appropriate and enable students to master concepts and skills as well as achieve learning outcomes.	4.67	0.93
10	The presented learning activity can stimulate students' curiosity.	4.83	0.97
11	Activities engage students and encourage cooperative learning in groups.	4.83	0.97
12	The activities in sub-topic 5.1 can help students distinguish and relate the physical and chemical properties and appearance of the three states of matter in terms of the arrangement and movement of particles.	4.67	0.93
13	The activities in sub-topic 5.2 can help students describe the changes in the state of matter in terms of the movement of particles as a result of the absorption and release of heat based on kinetic theory. Students will explain (with examples) the changes in the state of matter in relation to daily life.	4.67	0.93
14	The activities can make students aware of the importance of problem-solving and building new knowledge.	4.50	0.90
15	Activities can make students aware of the value of science in daily life and science learning.	4.50	0.90
16	The activity is suitable for high- and low-level ability students.	4.67	0.93
17	The activity emphasises the construction of concepts set out in the curriculum.	4.50	0.90
Computational Thinking Skills			
18	The sequence of introducing computational thinking skills is appropriate.	4.50	0.90

19	Activities can cultivate computational thinking skills.	4.67	0.93
20	Activities can cultivate students to be critical thinkers, creative and innovative.	4.67	0.90
21	Activities can help students in solving problems systematically related to the Matter topic.	4.50	0.90
22	Integrating computational thinking in teaching and learning promotes Higher Order Thinking Skills (HOTS)	4.50	0.90
23	The activities can cultivate CT skills		
	a. <i>Algorithm</i>	4.50	0.90
	b. <i>Decomposition</i>	4.50	0.90
	c. <i>Abstraction</i>	4.50	0.90
	d. <i>Generalisation</i>	4.67	0.93
	e. <i>Evaluation</i>	4.67	0.93

The experts were also asked to answer an open-ended question about the module, and it can be summarised as follow:

- *The content was simple, concise, and compact. It makes it very easy for students to understand basic concepts and facts.*
- *The Science module facilitates students to make references and revisions.*
- *Very effective in improving students' understanding of concepts and facts about the matter.*
- *The content in the questions is relevant to the researcher's study and appropriate to the learning objectives.*

Besides that, the experts also were asked to evaluate the language used in the module. Table 10 shows the analysis of language assessment of the module in percentage (%). The analysis findings showed that all items have a high "Yes" response (100%) except for items seven and eight, with a response of 90% and 80%, respectively. Therefore, based on the analysis, it could be said that the module is valid and is suitable to be implemented with the actual students.

Table 10:

Analysis of the Response Percentage for Each Item in the Module Language Assessment Questionnaire

No.	Item	Response	
		Yes (%)	No (%)
1	Each sentence gives a clear meaning	100.0	0.0
2	The language used is easy to understand	100.0	0.0
3	The instructions given are clear	100.0	0.0
4	The labels used are appropriate	100.0	0.0
5	The font size is appropriate and easy to read	100.0	0.0
6	Writing spacing is appropriate	100.0	0.0
7	The terms used are precise	90.0	10.0
8	Concepts on Science/Computer Science used is appropriate	80.0	20.0
9	The objectives of the module are clearly stated	100.0	0.0
10	There are no spelling mistakes	100.0	0.0

Conclusion

This study emphasises the design and development of the Matter module based on the five phases in the ADDIE model, namely Analysis, Design, Development, Implementation, and Evaluation. The findings from each phase were interrelated. It can be concluded that the generic ADDIE model with cooperative and project-based learning strategy has been proven to be significant in the systematic development of the Matter module that integrates computational thinking skills. The formative evaluation from the experts indicates that the module's content is suitable for teaching Matter topics, and the learning activities designed in the module can be said to be valid to promote computational thinking skills. For further study, the module may be tested with the actual students to evaluate the effectiveness.

References

- Abd Razak, A. Z., Surat, S., & Abd Majid, R. (2020). The Design of GIFTED Motivation Module Using the ADDIE Model Approaches Among the Gifted and Talented Students. *International Journal of Academic Research in Progressive Education and Development*, 9(2). <https://doi.org/10.6007/ijarped/v9-i2/7494>
- Azmi, R. D., & Ummah, S. K. (2021). Implementasi Project Based Learning Untuk Mengetahui Kemampuan Computational Thingking Mahasiswa. *Jurnal Ilmiah Pendidikan ...*, 5(1).
- Bicer, A., Capraro, R. M., Capraro, M. M., Oner, T., & Boedeker, P. (2015). STEM Schools vs. Non-STEM Schools: Comparing Students' Mathematics Growth Rate on High-Stakes Test Performance. *International Journal on New Trends in Education and Their Implications*, 6(1).
- Chongo, S., Osman, K., & Nayan, N. A. (2021). Impact of the Plugged-in and Unplugged Chemistry Computational Thinking Modules on Achievement in Chemistry. *Eurasia Journal of Mathematics, Science and Technology Education*, 17(4). <https://doi.org/10.29333/ejmste/10789>
- Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). Computational thinking A guide for teachers. *Computing At School*, October 2018.
- Khasyyatillah, I., & Osman, K. (2019). The Development of CT-S Learning Module on The Linear Motion Topic to Promote Computational Thinking Thinking. *Journal of Educational Sciences*. <https://doi.org/10.31258/jes.3.3.p.270-280>
- Lee, S. W., & Khalid, F. (2018). Primary School Students' Experiences during Game Development Project Using Scratch. *International Journal of Engineering & Technology*, 7(4.21), 16–19.
- Liu, F., & Yang, K. (2021). Exploration on the teaching mode of contemporary art computer aided design centered on creativity. *Computer-Aided Design and Applications*, 19(s1). <https://doi.org/10.14733/CADAPS.2022.S1.105-116>
- Rahayu, T., & Osman, K. (2019). Knowledge Level and Self-Confidence on The Computational Thinking Skills Among Science Teacher Candidates. *Jurnal Ilmiah Pendidikan Fisika Al-Biruni*. <https://doi.org/10.24042/jipfalbiruni.v8i1.4450>
- Seawright, L. L. (2003). Reducing Learning Object Inspection/Evaluation Costs in Instructional Design. In *ProQuest Dissertations and Theses*.
- Selby, C. C., & Woollard, J. (2014). Refining an Understanding of Computational Thinking. *Author's Original*.
- Senin, S., & Nasri, N. M. (2019). Teachers' Concern towards Applying Computational Thinking Skills in Teaching and Learning. *International Journal of Academic Research in Business and Social Sciences*. <https://doi.org/10.6007/ijarbs/v9-i1/5398>

- Shin, N., Bowers, J., Krajcik, J., & Damelin, D. (2021). Promoting computational thinking through project-based learning. *Disciplinary and Interdisciplinary Science Education Research*, 3(1). <https://doi.org/10.1186/s43031-021-00033-y>
- Sidek, M. N., & Jamaludin, A. (2017). PEMBINAAN MODUL Bagaimana Membina Modul Latihan dan Modul Akademik. Serdang: Universiti Putra Malaysia.
- Wilson, B., Jonassen, D., & Cole, P. (1993). Cognitive Approaches to Instructional Design. *The ASTD Handbook of Instructional Technology*, 3.
- Wan Hanim Nadrah Wan Muda. 2017. Kesan Kaedah Pembelajaran Visual Dalam Usaha Meningkatkan Kefahaman Pelajar Bagi Subjek Matematik II. Fakulti Pendidikan Teknikal Dan Vokasional. Universiti Tun Hussien Onn.
- Wing, J. M. (2006). Computational Thinking. *Communications of the Association for Computing Machinery (ACM)*, 49(3), 33–35.
<https://doi.org/https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf>