DOI: 10.15388/infedu.2025.27

GenAI-Assisted Data Science Course to Promote GenAI Literacy for Non-Computing Students

Oscar KARNALIM*, Hapnes TOBA, PETER,
Jahja Hamdani WIDJAJA, Tan Ming KUANG, David Try LIPUTRA,
Vivi ARISANDHY

Maranatha Christian University, Surya Sumantri Street No. 65, Indonesia
e-mail: {oscar.karnalim; hapnestoba}@it.maranatha.edu,
{peter; jahja.hamdani; tm.kuang}@eco.maranatha.edu,
{david.tl; vivi.arisandhy}@eng.maranatha.edu

Received: February 2025

Abstract. Given the emergence of GenAI, students should develop GenAI literacy to promote its benefits and mitigate its drawbacks. However, many studies focus on enhancing their learning experience with GenAI, not understanding GenAI literacy. Two studies are dedicated to GenAI literacy, but they either require additional sessions or focus on overly specific tasks. We integrate GenAI into a data science course and its assessments to specifically promote GenAI literacy for non-computing students. The course design expects students to learn from their direct experience with GenAI, especially regarding GenAI usability, reliability, ethics, and privacy. Students are encouraged to use and acknowledge GenAI for some assessments and to align GenAI-generated programs to their own styles. Our evaluation involving 113 students showed that the course design might help students to understand GenAI characteristics and change their behaviour. Students are unlikely to be involved in GenAI misuse. Further, they align GenAI-generated programs and acknowledge their use. From the educational viewpoint, students could also achieve the course learning objectives.

Keywords: GenAI assistance, data science, GenAI literacy, learning technology, programming.

1. Introduction

Generative Artificial Intelligence (GenAI) is a form of AI that focuses on generating content from large datasets (Jovanovic and Campbell, 2022). The use of GenAI in education has become increasingly inevitable. Since the emergence of publicly accessible large language models (LLMs) in 2023, in particular ChatGPT, an increasing number of students have been utilising generative AI (GenAI) to complete their class assignments

---

* Corresponding author.

(Niloy *et al.*, 2024; Ngo, 2023). One of the most significant advantages of GenAI is that it increases effectiveness and accelerates responses to questions by providing direct access to results (Shi *et al.*, 2025; Spatharioti *et al.*, 2023).

GenAI can help students in learning including increasing productivity (Hung and Chen, 2023) and explaining learning materials (Jury *et al.*, 2024). Despite the benefits, GenAI introduces several drawbacks. Students might over-rely on GenAI, negatively affecting their cognitive skills (Zhai *et al.*, 2024). Another major concern is GenAI-assisted plagiarism, where students with GenAI can complete assessments with limited understanding of the subject matter (Karnalim *et al.*, 2024b).

Students should be aware of GenAI literacy to promote GenAI benefits and mitigate the drawbacks. Although there is no consensus on the definition of GenAI literacy (Almatrafi *et al.*, 2024), it typically refers to the ability to evaluate GenAI critically and collaborate effectively with it (Long and Magerko, 2020). There are many studies discussing GenAI in education, but most of them are about reviews (Ng *et al.*, 2021; Yusuf *et al.*, 2024), and/or recommendations (Chiu, 2024; Luo, 2024).

Several studies are focused on integrating GenAI into student learning. However, in most cases, the goal is to promote student understanding of the subject matter (Llerena-Izquierdo *et al.*, 2024; Güner and Er, 2025; Cubillos *et al.*, 2025). It does not specifically cover GenAI literacy. Two studies focus on GenAI literacy. Tzirides *et al.* (2024) allow 37 students to interact with two types of GenAI: reviewer and image generator. Students are more comfortable with GenAI and can critically assess GenAI applications. This study focuses on overly-specific tasks, which makes it less practical. Korte *et al.* (2024) design five hours of online workshops to promote GenAI literacy for 29 students. The students become more aware of GenAI and its importance. The study needs additional sessions to be added to existing courses.

In response to the aforementioned gaps (overly-specific tasks and additional sessions), we present a GenAI integration that is fully applied in a course and its assessments, eliminating the need for additional sessions. Further, it can be generalised to other courses. In our case, the integration is conducted in a data science course, which is becoming increasingly popular among non-computing students (Liu *et al.*, 2023). The integration tailors student experiences in GenAI usability, reliability, ethics, and privacy. Students are encouraged to use and acknowledge GenAI for some assessments (but not all to prevent over-reliance). They are also expected to align GenAI results to their own styles to showcase their understanding. Our similarity detector might detect and mark directly copied GenAI results, as they share high similarity (Pang and Vahid, 2024). To the best of our knowledge, the integration is the first of its type. The data science course was offered to 113 non-computing undergraduate students (management) for a semester.

To assess the effectiveness of our GenAI-assisted data science course, the study has a number of research questions:

- **RQ1** Does the GenAI-assisted data science course help students to understand GenAI literacy?
- **RQ2** Do students align GenAI results to their own needs and styles?
- **RQ3** Do students achieve the data science's learning objectives?

Our GenAI-assisted data science course can be an alternative for non-computing disciplines to integrate data science into their curriculum. GenAI assistance reduces the complexity of data science for non-computing students. They do not need to write the programs from scratch and can easily ask for help. Further, it is also useful for promoting GenAI literacy, especially regarding usability, reliability, privacy, and ethics.

2. Method

2.1. *GenAI-Assisted Data Science Course*

Our GenAI integration aims to promote GenAI literacy by leveraging student experiences with GenAI, focusing on usability, reliability, privacy, and ethics. Students can use GenAI to learn course materials or complete assessments. However, they must acknowledge the use of GenAI and align GenAI results with their own needs and styles. To prevent overreliance, students are expected to complete some assessments independently.

To integrate our method, instructors should follow four steps described in Fig. 1. First, instructors should set their expectations about GenAI and ethics. They should also consider how to prevent students from cheating. The expectations will be the baseline for designing all assessments. Second, four GenAI literacy aspects (usability, reliability, privacy, and ethics) should be integrated into existing assessments. For integrating GenAI ethics, some assessments should allow the use of GenAI, but it should be acknowledged and aligned with their own needs. Other aspects should be integrated by at least one assessment. Usability should encourage students to use GenAI for a particular task. Reliability should discuss the limitations of GenAI for providing factual data. Privacy should focus on showing that GenAI uses accessible data. Third, there should be assessments that can evaluate student competence without the use of GenAI. They are preferred to share the same topic as assessments with GenAI. Finally, instructors should design the introductory session to describe their expectations regarding GenAI and ethics. The session can include some real-life examples.

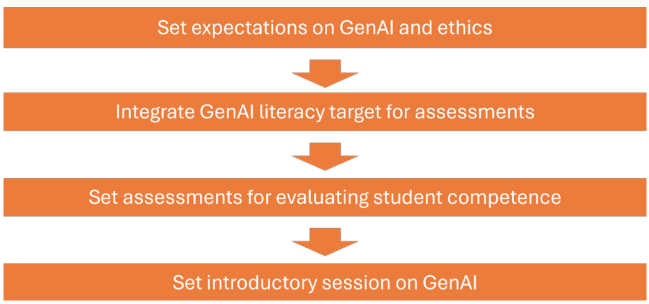

Fig. 1. Steps to integrate our method.

The integration fulfils four aspects of core competencies in AI literacy as defined by Long and Magerko (2020) : knowing, understanding, using, and evaluating. By allowing students to use GenAI and informing them about the characteristics, they are expected to know, understand, and use GenAI. As GenAI results need to be aligned to the students' needs and styles, they are expected to be able to evaluate GenAI.

It also covers all three dimensions of AI literacy as defined by Ng *et al.* (2021). The knowledge dimension is achieved by informing students about GenAI. The skills dimension is achieved by allowing students to use GenAI while completing some assessments. The ethics dimension is achieved by promoting critical behaviour toward GenAI results.

In our case, the integration is applied to a data science course, which is becoming popular for non-computing students. The course focuses on introducing data science concepts with a little bit of programming.

The GenAI-assisted data science course spans 16 weeks with 14 class meetings, one mid-test, and one final test. The syllabus details can be seen in Table 1. Week 01 and 02 focus on concepts of data science and basic Python programming. Week 03 to 06 focus on data visualisation and its Python implementation. Week 07 focuses on communicating the findings from data visualisation and progress for the mid-test. Week 08 focuses on the mid-test, which is a group project presentation. Details of the projects are provided in Week 06. Students are expected to search publicly available data related to their own discipline (management in our case), generate several data visualisations in Python, and communicate the findings. GenAI assistance is allowed but needs to be acknowledged.

Table 1

GenAI-assisted data science course syllabus

Week	Theoretical topic	Technical topic
01	Data science and GenAI introduction	Python comments, variables, operators, input-output, if-else
02	Data quality	Python loops and functions
03	Data visualisation, bar chart, line chart, and pie chart	Python implementation of the visualisations
04	Scatter plot, histogram, heat map, and box plot	Python implementation of the visualisations
05	Area chart, bubble chart, tree map, violin plot	Python implementation of the visualisations
06	Word cloud, 3D surface plot, network graph, Sankey diagram	Python implementation of the visualisations
07	Communicating data findings	Mid-test progress
08	Mid-test	-
09	Business analytics framework, introduction to machine learning, and linear regression	Python linear regression
10	Business intelligence, big data, and logistic regression	Python logistic regression
11	Naive Bayes and K-nearest neighbour	Python Naive Bayes and K-nearest neighbour
12	Multiple regressions and support vector machine	Python multiple regressions
13	K-means clustering and decision tree	Python K-means clustering
14	Case study: machine learning classification part 1	Python implementation of the case study
15	Case study: machine learning classification part 1	Python implementation of the case study
16	Final test	-

Week 09 and 10 discuss business data science and the basics of machine learning. Week 11 to 13 discuss more machine learning algorithms and implementations. Week 14 and 15 focus on a case study to implement a machine learning classification. Week 16 concerns the final test: a group project presentation. Students must compare several machine learning models for data classification and analyse the findings. The data should mix public, artificial, and student-owned data. GenAI assistance is allowed but must be acknowledged.

Each class meeting is typically scheduled as follows:

1. Theoretical lecture (60 mins).
2. Technical review assessment (30 mins).
3. Technical lecture (30 mins).
4. Supervision of technical homework assessment (30 mins).

Theoretical lectures focus more on data science concepts and GenAI introduction. During the lectures, students are encouraged to ask GenAI to clarify several learning materials. GenAI introduction is conducted in the first week. Students are informed about GenAI with ChatGPT as the main focus. The instructors tell a story about an instructor's experience, asking ChatGPT about themselves. The responses vary but improve over time as long as the expected responses are provided. Students could understand that GenAI responses might be incorrect and/or biased (reliability). They will also be informed about two things: 1) how GenAI collects and uses data (privacy) and 2) instructors' expectations of academic integrity (ethics). Finally, there will be a demonstration of GenAI prompt engineering, especially showing students how to generate and assess code for programming assessments (usability).

Technical lectures focus on Python implementation of some topics discussed in the theoretical lectures. The implementation employs several built-in libraries, including matplotlib, seaborn, and scikit-learn. To promote usability, students are encouraged to interact with GenAI during the learning process.

Technical homework assessments test students' knowledge about technical lectures in the corresponding weeks. The assessments should be completed individually before the next class meeting. They will be issued each week with a 30-minute supervision session with the instructors. Assessment 07, 14, and 15 were the only ones that should be completed in groups, and the marks would be part of either the mid-test (Assessment 07) or the final test (Assessment 14 and 15). Only one student in each group was expected to submit the work. The group formation was similar to the mid-test and the final test. The homework assessments are specifically designed to promote GenAI literacy. For each assessment, they are expected to use GenAI to generate code and/or fix errors. Table 2 shows the assessments and their primary GenAI literacy aspects. All GenAI literacy aspects are covered in Assessment 07, 12, 14, and 15. Usability focuses mainly on Assessment 01, 02, and 10. The assessments explicitly encourage students to use GenAI to obtain inspiration or even programming code. Reliability focuses mainly on Assessment 03–06, and 11. The assessments typically ask students to obtain factual data from GenAI and then validate it. Privacy focuses mainly on Assessment 09 and 13. The assessments typically ask students to obtain personal data of public figures from GenAI. GenAI eth-

Table 2
Technical homework assessments

Week	Task	Primary GenAI literacy
01	Introductory problem solving: currency conversion	Usability
02	Introductory problem solving: number sequences	Usability
03	Implementing selected visualisations with data searched with GenAI but validated by the students	Reliability
04	Implementing selected visualisations with data searched with GenAI but validated by the students	Reliability
05	Implementing selected visualisations with data searched with GenAI but validated by the students	Reliability
06	Implementing selected visualisations with data searched with GenAI but validated by the students	Reliability
07	Mid-test preparation	All
09	Linear regression for influencer data	Privacy
10	Case study for logistic regression	Usability
11	Naive Bayes and K-Nearest Neighbour with GenAI-generated data	Reliability
12	Multiple regressions for the stock market with GenAI assistance on feature selection and data collection	All
13	K-means clustering with student data	Privacy
14	Case study developing a machine learning classification	All
15	Case study developing a machine learning classification	All

ics are applied to all assessments by acknowledging any GenAI help and not directly reusing GenAI results.

Technical review assessments are similar to the technical homework assessments from previous weeks. For example, in Week 03, the technical review assessment will cover Week 02's technical homework assessment. Nevertheless, the scope is more limited, as the assessments should be completed in 30 minutes. Further, the task is slightly aligned so that students cannot just memorise the solution to last week's technical homework assessment. Technical review assessments are given in Week 02–07, and Week 10–13.

While completing the technical review assessments, students can neither use GenAI nor access last week's resources, ask their colleagues, or ask the instructors. They should complete the assessments by themselves, showcasing their understanding of the course materials and reducing student over-reliance on GenAI. Since the course is offered to non-computing students, template code is given for each technical review assessment to reduce the complexity of the assessment.

To promote ethics, all weekly assessments are submitted to our own assessment submission system that immediately reports obvious program similarity (Karnalim *et al.*, 2024a). Plagiarised programs tend to be similar, though high similarity is not always a result of plagiarism (Simon *et al.*, 2020). Further, on courses allowing GenAI use, directly copied GenAI-generated programs are more likely to share high similarity to one another (Pang and Vahid, 2024). Students will be notified if their GenAI-assisted programs are less likely to align with their programming styles.

Once the assessment closing time passes, all submissions will be checked for plagiarism and GenAI misuse (i.e., direct copying of GenAI results) by comparing them against one another. Any misconduct will be penalised by assigning zero marks to all involved students.

According to Bloom's taxonomy (Momen *et al.*, 2022), our course design aims to help students remember, understand, and apply GenAI knowledge (the first three levels). The goals are similar for data science knowledge, except that the course also aims to develop analysis skills (level 4).

While our study is focused on a data science course, our GenAI integration can be applied to other courses as well. The courses should allow the use of GenAI with proper acknowledgement. Further, the assessments should cover at least four aspects of GenAI literacy: usability, reliability, privacy, and ethics. To prevent over-reliance, the use of GenAI can be restricted in some assessments.

2.2. *Addressing the Research Questions*

RQ1 concerned whether the GenAI-assisted data science course helps students understand GenAI literacy. It was addressed by two post-questionnaire surveys (general GenAI literacy survey and ethics survey), a summary of the acknowledgement sections, and the numbers of students involved in plagiarism and GenAI misuse.

The general GenAI literacy survey was developed through consensus among the authors based on the literature review and the institution's GenAI policies. The surveys were then expanded and validated by three course instructors based on their expectations of GenAI and its ethical implications. The survey covered four aspects of GenAI: usability, reliability, privacy, and ethics. Each question should be responded to on a 5-point Likert scale where 1 refers to 'strongly disagree', 3 refers to 'neutral', and 5 refers to 'strongly agree'. Details of the questions could be seen in Table 3. Most questions covered usability. G10 checked whether students see the benefits of GenAI. If so, they were likely to use it for their future learning. G11–G14 were the detailed versions of G05. G15–G18 are the detailed versions of G06. G21 measured the effectiveness of our GenAI introduction session in the first week's meeting.

The GenAI literacy survey's results would be analysed by averaging the Likert scores. For each question, a higher score was preferred. Any questions with overly high or overly low scores would be discussed. We did not use the existing GenAI literacy survey, as they were quite general. We wanted to specifically address the effectiveness of GenAI integration in data science assessments.

A Structural Equation Modelling (Ullman and Bentler, 2012) would be applied based on grouping in Table 3. All categories, along with G21, were then directed to the general GenAI literacy score (which is the average of all scores). G11–G14 were not included because their purpose was to identify a task where GenAI was most helpful in generating code. The same applied to G15–G18, but for fixing errors. A reliability test would be conducted on the survey responses using Cronbach's alpha (Cronbach, 1951).

Table 3
General GenAI Literacy Survey

ID	Question	Primary GenAI literacy
G01	GenAI usage promotes awareness of GenAI	Usability
G02	GenAI is useful to search ideas for case studies	Usability
G03	GenAI is useful to search data for analysis	Reliability
G04	GenAI is useful to generate sample data for analysis	Reliability
G05	GenAI is useful to generate program code	Usability
G06	GenAI is useful to fix code errors	Usability
G07	GenAI need manual verification to validate their responses	Reliability
G08	GenAI can generate unreliable responses	Reliability
G09	GenAI might use user data to generate solutions for other users	Privacy
G10	I will use GenAI for my future learning	Usability
G11	GenAI is useful to generate program code for data visualisation	Usability
G12	GenAI is useful to generate program code for machine learning classification	Usability
G13	GenAI is useful to generate program code for machine learning regression	Usability
G14	GenAI is useful to generate program code for machine learning clustering	Usability
G15	GenAI is useful to fix code errors for data visualisation	Usability
G16	GenAI is useful to fix code errors for machine learning classification	Usability
G17	GenAI is useful to fix code errors for machine learning regression	Usability
G18	GenAI is useful to fix code errors for machine learning clustering	Usability
G19	Any GenAI help should be acknowledged	Ethics
G20	Any unacknowledged GenAI help is considered plagiarism	Ethics
G21	GenAI introduction provides me basic knowledge about GenAI	All
G22	Any GenAI results should be understood before being used	Ethics
G23	Any GenAI results should be aligned to one's work if used	Ethics

As stated before, G11–G18 were designed to identify a task where GenAI was most helpful in generating code and fixing errors. Any differences would be tested with a one-way ANOVA (Ross and Willson, 2017) while their effect sizes would be measured with Omega squared.

The ethics survey was dedicated to ethics in the era of GenAI and adapted from another study (Simon *et al.*, 2014). The survey asked students' perceptions regarding a number of scenarios, and whether they are academically acceptable. Details of the scenarios could be seen in Table 4. Each scenario should be responded by either "academically acceptable", "academically unacceptable", or "do not know". E01–E11 covered conventional ethics, which might be affected by our course design's GenAI practice. E03–E11 were about plagiarism and collusion (Simon *et al.*, 2014). E01 and E02 were about contract cheating, asking someone to complete one's own work (Manoharan and Speidel, 2020). E12–E20 were the GenAI version of E02 and E04–E11. A colleague's assistance was replaced with GenAI assistance. Most scenarios were deemed academically acceptable, except E12, E13, and E19.

Table 4
Ethics Literacy Survey

ID	Question	Expectation
E01	Purchasing code written by other students to incorporate into your own work	unacceptable
E02	Paying another student to write the code and submitting it as your own work	unacceptable
E03	Basing an assessment largely on work that you wrote and submitted for a previous course, without acknowledging this	unacceptable
E04	Incorporating the work of another student without their permission	unacceptable
E05	Copying another student's code and changing it so that it looks quite different	unacceptable
E06	Copying an early draft of another student's work and developing it into your own	unacceptable
E07	Discussing with another student how to approach a task and what resources to use, then developing the solution independently	acceptable
E08	Discussing the details of your code with another student while working on it	acceptable
E09	Showing troublesome code to another student and asking them for advice on how to fix it	acceptable
E10	Asking another student to take troublesome code and get it working	unacceptable
E11	After completing an assessment, adding features that you noticed when looking at another student's work	acceptable
E12	Subscribing to GenAI service to generate the code and submitting it as your own work	unacceptable
E13	Incorporating work from GenAI service without acknowledgement	unacceptable
E14	Copying code generated by GenAI service and changing it so that it looks quite different	acceptable
E15	Copying an early draft of the solution generated by GenAI service and developing it into your own	acceptable
E16	Asking to GenAI service how to approach a task and what resources to use, then developing the solution independently	acceptable
E17	Discussing the details of your code with GenAI service while working on it	acceptable
E18	Showing troublesome code to GenAI service and asking it for advice on how to fix it	acceptable
E19	Asking GenAI service to take troublesome code and get it working	unacceptable
E20	After completing an assessment, adding features that you were inspired from GenAI service	acceptable

Students' ethics literacy would be measured based on the proportion of correct responses. A higher proportion was preferred, showing students' high awareness of the topic. Any interesting findings would be discussed.

For both GenAI and conventional ethics surveys, a reliability test would be conducted on the responses using Cronbach's alpha (Cronbach, 1951). Further, responses for GenAI questions would be compared to those of its conventional counterparts with a one-way ANOVA (Ross and Willson, 2017), while its effect size was measured using Omega-squared.

For all surveys, the authors would manually check whether each response was completed haphazardly based on two factors: completion time and the question responses. If a student completed the survey in less than a minute, it was likely that they did not read the questions. The same thing applied when all questions were answered with the same answer (either the most right, the middle, or the most left option). Other survey-related statistical analyses were not applicable, as all responses came from the same course offering.

Students who were aware of ethics might be less likely to be involved in plagiarism or GenAI misuse. Hence, the number of students involved in plagiarism and GenAI misuse would also be reported for further analysis. Fewer suspected submissions were preferred.

Descriptive statistics and thematic analysis of the acknowledgement sections of technical homework assessments, the mid-test, and the final test would also be provided to further address RQ1. Students were expected to acknowledge any GenAI assistance on their assessments. The summary would show the general use of GenAI for students in completing assessments.

RQ2 was about whether students aligned GenAI-generated programs to their own styles. It was addressed by reporting the proportion of highly similar program pairs compared to all possible submission pairs on the technical homework assessments. A smaller proportion showed that students aligned GenAI-generated programs with their own style. Each student's submission might have its own style. The reporting was aided by our similarity detector (Karnalim, 2023). It compared student submissions to one another with Super-Bit, a locality-sensitive hashing algorithm (Ji *et al.*, 2012). Any submission pairs with at least half of the similar content will be reported. The proportion of highly similar programs was measured based on the total number of submission pairs. Assessments with the highest or the lowest proportions would be specifically discussed.

RQ3 was about whether students achieve the learning objectives of the data science course. It would be addressed based on student performance. High performance was preferred.

3. Results and Discussion

3.1. *General GenAI Literacy Understanding (RQ1)*

As shown in Fig. 2 and Fig. 3, students generally saw the benefits of GenAI. Further, their responses on our GenAI literacy survey were reliable, as indicated by Cronbach's alpha. The internal consistency score was 0.94. Moreover, no students completed the survey hapzardly.

Fig. 2 showed that students understood the benefits of GenAI regarding its usability, though the extent varied. This was expected since our course design specifically encouraged students to use GenAI. They might learn how to use it effectively and thus complete assessments more efficiently.

Students clearly saw the benefits of GenAI in finding ideas for case studies (G02). Our course design required students to have their own case studies for both the mid and final tests, based on their discussions with GenAI. There were also some assessments focusing on case studies (Week 7, 10, 14, and 15).

They also believed that GenAI was particularly useful in generating code for data visualisation (G11), though the difference compared to other tasks (G12–G14) were not statistically significant according to a one-way ANOVA. The f-statistic was 1.2, while

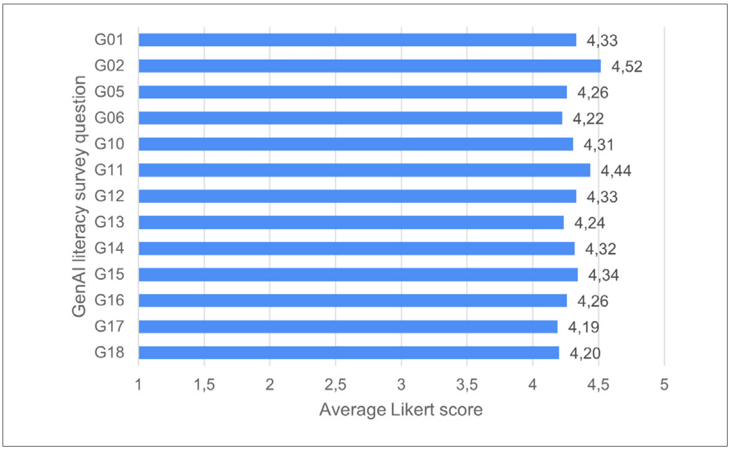

Fig. 2. Likert score about GenAI usability from GenAI literacy survey.

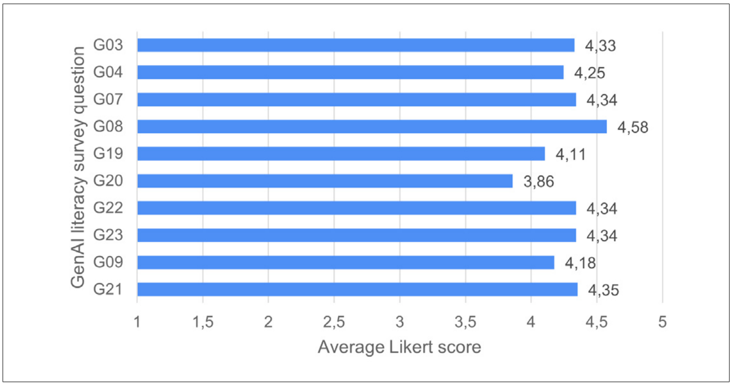

Fig. 3. Likert score about GenAI reliability (G03, G04, G07, G08), ethics (G19, G20, G22, G23), privacy (G09), and introduction session (G21) from GenAI literacy survey.

the p-value was 0.3 with a small Omega-squared effect size (less than 0.01). Assessments asking for data visualisation were quite simple, and the code was straightforward. The generated code was a bit simpler than that for machine learning tasks (classification, regression, and clustering). Code for machine learning tasks could sometimes use advanced syntax, such as Pandas' data frame and/or list comprehension.

Although students perceived GenAI as useful, its effectiveness was a bit lower in fixing programming errors for machine learning regression and clustering (G17 and G18). There were only two assessments discussing those topics, which is far fewer than for assessments on data visualisation and machine learning classification. Students might be better at using GenAI for particular tasks if encouraged to complete those tasks frequently. It is worth noting that, according to a one-way ANOVA, the differences were not statistically significant. The f-statistic was 0.79, the p-value was 0.49, and the Omega-squared effect size was less than 0.01.

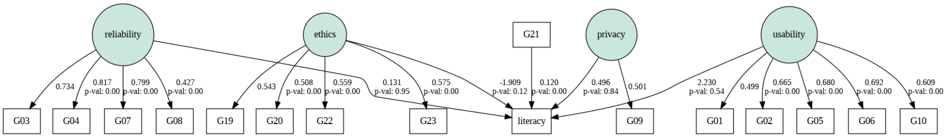

Fig. 4. General GenAI literacy model.

Students were generally aware of GenAI reliability, as shown in Fig. 3. The course design provided students with firsthand experience, helping them understand that GenAI could still be useful for searching and generating data (G03 and G04). However, they need to validate the responses (G07) as some responses could be unreliable (G08).

Regarding ethics, students seemed to understand that GenAI help should be acknowledged (G19). Otherwise, it is considered plagiarism (G20). Further, it was recommended to understand any GenAI responses before incorporating them into students' work (G22). To show the understanding, GenAI responses should be aligned to students' needs and writing (G23). Instructors should discuss more with students that unacknowledged GenAI use was considered plagiarism (G20). Students might think that small portions of GenAI results did not need explicit acknowledgement.

Regarding privacy, students understood that data fed to GenAI might be used to generate solutions for other users (G09). This knowledge was fundamental, so students did not accidentally give sensitive data to GenAI.

The introduction session with a demonstration helped students to understand GenAI (G21). Such a session was important since not all non-computing students had used GenAI.

The general GenAI literacy survey was mapped into a model shown in Fig. 4. The chi-square test was 249.4 with degrees of freedom of 96 and a p-value less than 0.001. Our model captured most of the structure, though the deviation was substantial. The ratio between the chi-square test statistic and the degrees of freedom was around 2.6, indicating a moderate fit. The moderate fit was also suggested by the Comparative Fit Index of 0.8, the Normed Fit Index of 0.73, and the Root Mean Square Error of Approximation of 0.138. The model, however, seemed too complex as its Tucker-Lewis Index was 0.76. Further, it did not explain the variance as expected. Its Goodness of Fit Index was 0.73, and its Adjusted Goodness of Fit Index was 0.66.

3.2. *Ethics Understanding (RQ1)*

Students generally understood conventional ethics in the era of GenAI, with an average correctness rate of 69%. Although the reliability of the responses was low (with Cronbach's alpha at 0.50), no students were identified to hapzardly completed the survey. Fig. 5 showed that students were particularly aware that it was preferred not to reuse one's work without permission (E04), not to pay another student to complete an assessment (E02), and to discuss how to solve assessment tasks so long as they completed the

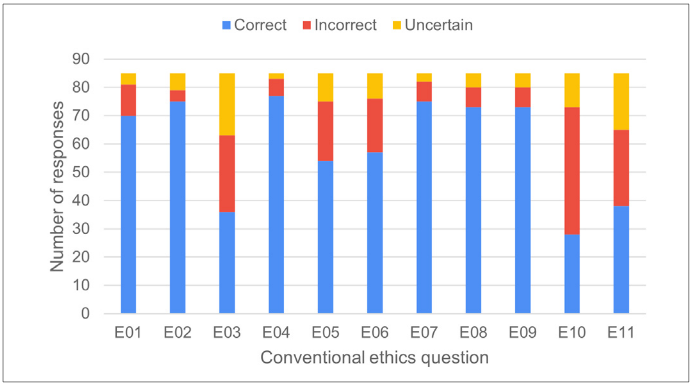

Fig. 5. Student responses to conventional ethics questions from ethics survey.

assessments independently (E07). Such understanding might be promoted as our course design encouraged students to acknowledge any help, to interact with GenAI, and not to directly reuse GenAI-generated solutions. It might also be a result of discussions among instructors and students involved in the course.

Students were not quite aware that instructors expected them not to ask a colleague to fix errors (E10) and not to expand their own work and then resubmit for another assessment (E03). However, they were allowed to draw inspiration from a colleague's work (E11). Our course design encouraged acknowledging any help, but at a general level. Students might then focus more on substantial help that affects a larger part of the solution. Further, it did not specifically cover expanding one's own work. Moreover, some students might consider inspiration from a colleague as a major help and thus need to be acknowledged, even though it was at a high level.

Students were slightly less aware of GenAI ethics compared to the general one, with a reliability score of 0.42 according to Cronbach's alpha. The average correctness rate was 63%, six percents lower than that of the general ethics. The difference was statistically significant, though the effect size was small, as shown in Table 5 with f-statistic

Table 5

Comparison between conventional and GenAI ethics with a one-way ANOVA

Conventional	GenAI	Difference	F-statistic	p-value	Effect Size	Significant
E02	E12	35.29%	29.64	< 0.001	0.14	significant
E04	E13	27.06%	19.41	< 0.001	0.09	significant
E05	E14	25.88%	12.06	< 0.001	0.06	significant
E06	E15	17.65%	5.55	0.01	0.02	significant
E07	E16	4.71%	0.77	0.38	< 0.01	insignificant
E08	E17	4.71%	0.67	0.41	< 0.01	insignificant
E09	E18	9.41%	2.47	0.11	< 0.01	insignificant
E10	E19	2.35%	0.1	0.74	< 0.01	insignificant
E11	E20	24.71%	11.15	0.001	0.05	significant
all	all	6%	10.7	0.001	0.05	significant

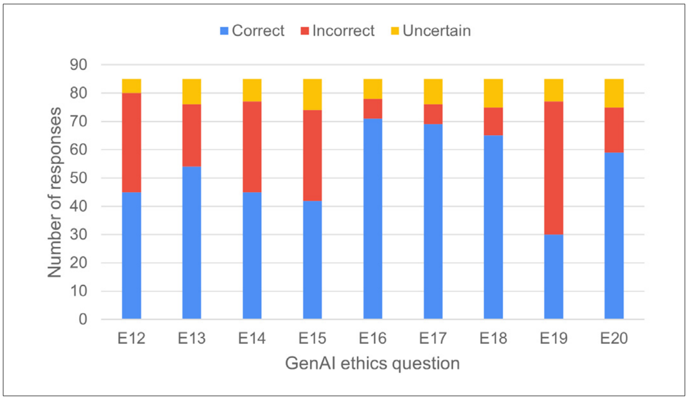

Fig. 6. Student responses to GenAI ethics questions from ethics survey.

of 10.7 and p-value 0.001. This was expected as GenAI ethics was relatively recent. Proportion of responses to GenAI ethics questions was shown in Fig. 6.

Students particularly understood that it was acceptable to ask GenAI how to approach a task and then develop the solutions independently (E16). It was also allowed to discuss their program code with GenAI (E17) and ask for recommendations (E18). E16 scenario often occurred while completing technical homework assessments, the mid-test, and the final test. GenAI results acted as the basis of their solutions. E17 and E18 scenarios occurred in most assessments, as students typically developed their programming solutions using GenAI.

Nevertheless, students did not really understand that asking GenAI to fix errors was unacceptable (E19). Unlike E18, E19 did not ensure that students understood the fix. Students might think E19 was acceptable because they might have accidentally done so while attempting to ask for recommendations.

It was interesting to see that only half of the students believed that expanding GenAI results to their own solutions was acceptable (E15). Further observation showed that some students tended to rewrite the whole solution as it was easier to align their own code. It was also the reason why E14 (copying GenAI results and then changing them) resulted in a relatively low correctness rate.

As our course design allowed students to use GenAI, some might think that subscribing to GenAI service to write the code was acceptable (E12). However, it was unacceptable since the generated code was not aligned with one's own writing. GenAI should assist rather than write the code directly.

When the responses to each GenAI ethics question were compared to its conventional counterparts, some of them resulted in statistically significant differences, though the effect size was small. Table 5 showed that the differences occurred on E12–E15 and E20. Student awareness of conventional ethics related to contract cheating (E02), plagiarism (E04–E06), and allowed inspiration from a third party (E11) did not necessarily result in similar awareness of GenAI ethics. Instructors needed to elaborate on these in detail during the introduction session. They should, in particular, explain why E14 and E15 were allowed while their counterparts (E05 and E06) were not allowed.

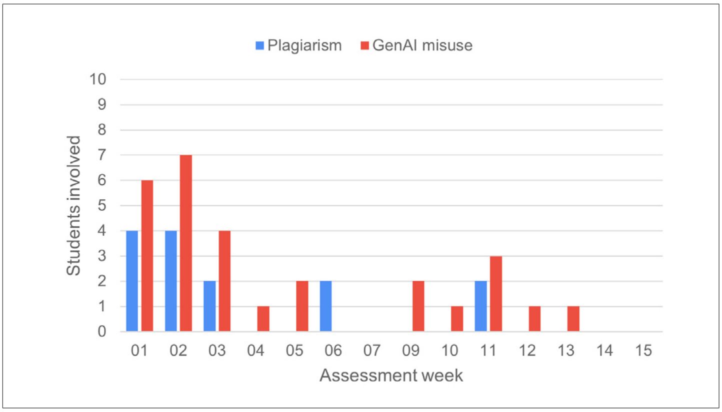

Fig. 7. Students involved in plagiarism and GenAI misuses on technical review assessments.

3.3. *Students Involved in Plagiarism and GenAI Misuse (RQ1)*

The number of students involved in plagiarism cases was relatively low. Fig. 7 showed that plagiarism only occurred on five of 14 assessments, with two or four students involved per assessment. Our course design imposed not only GenAI ethics but also the general ones. Further, since it required students to acknowledge GenAI help, it might help students to understand ethics in work ownership.

Regarding GenAI misuse, the number of involved students was slightly higher due to its recent emergence. However, it was still acceptable for a total of 113 students. Our course design promoted ethical and responsible use of GenAI by acknowledging and aligning the help. Students were discouraged from directly copying and pasting GenAI results. Assessment 01 and 02 reported many involved students in GenAI misuse. Perhaps they were still adapting to the course policy and programming. It was probably also a reason why more students were involved in plagiarism.

3.4. *Acknowledged GenAI Assistance (RQ1)*

For technical homework assessments, our study expected a total of 1582 student submissions, given that there were 113 students and 14 assessments. However, students occasionally failed to complete assessments, resulting in only 1417 submissions.

Based on these submissions, nearly three-fourths (73% or 1034 submissions) acknowledged GenAI usage in either program comments or separate text files. It might indicate that our course design effectively promoted GenAI ethics. However, some of the acknowledgements were overly general. There was a need to encourage students to provide more detail about how GenAI contributed to their solutions. More acknowledgement examples might be necessary for the GenAI introduction session.

Around half of the acknowledgements (572) were overly general, just mentioning GenAI services that they had used. More than a third (374) were more detailed, acknowledging GenAI services with their contribution to the assessments. For example, a program comment in a submission stated, "ChatGPT to draft code for visualising the bar chart". Our thematic analysis showed that these acknowledgements pertained to either code-writing assistance (231), error handling (87), or data searching (56). Code-writing assistance and error handling were expected to be common, since students had limited programming knowledge. Data searching was explicitly encouraged in some assessments.

The remaining acknowledgements (88) were very comprehensive. They not only mentioned the GenAI products with the contributions but also the steps and prompts employed to generate their own solutions with GenAI. All of them were about GenAI in helping the code-writing process. Around half of them (46) also reported GenAI in fixing errors. A third (29) covered more on GenAI in searching data and clarifying technical terminologies.

On the mid-test, 29 student groups were involved. Each group was expected to submit a report with a GenAI acknowledgement section. All groups did that, and the acknowledgements were quite comprehensive. Each acknowledgement could report more than one GenAI use. It was perhaps because the mid-test was a group project with a larger marking weight.

Our thematic analysis showed that 26 groups mainly used GenAI to assist their Python code-writing and error handling. During the first half of the course, students were still adapting to Python programming. They often asked GenAI about programming syntax and recommendations to fix errors. Seven groups reported the use of GenAI in analysing data. They asked GenAI to provide some implications of public data. Six groups searched for visualisation inspirations with GenAI. Given a set of data, GenAI was expected to provide suggestions on relevant visualisation.

In the final test, around two-thirds of the groups (18) still used GenAI to assist their code-writing process and error handling. It was lower than that of the mid-test. Further, they reported GenAI helps in code optimisation, which was not mentioned during the mid-test. During the final test, students became more fluent in programming and did not rely too much on GenAI for code-writing. GenAI contributed more to code optimisation.

Seventeen groups used GenAI to help their analysis, seeking inspiration for model selection, data preprocessing, model evaluation, and even result analysis. They focused more on clarifying terminologies and the impact of a particular selection.

3.5. *Aligned GenAI-Generated Programs to Students' Own Styles (RQ2)*

Students generally aligned their work to their own syntax and style. Fig. 8 showed that the proportion of similar submission pairs was no more than 5% for technical homework assessments. Most submissions did not share more than half of their content with other submissions. A high proportion might be a result of the complexity of the assessments. It introduced difficulty in understanding and modifying GenAI results. In Assessment 02, students just learned about programming loops and functions. They were quite complex for novices. The same happened for Assessment 05 and 06, covering complex visuali-

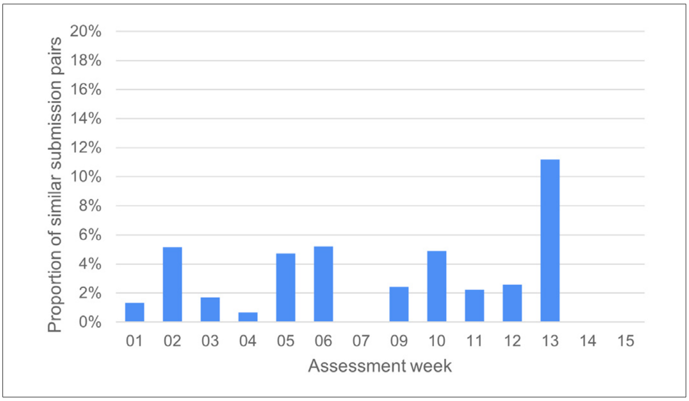

Fig. 8. Proportion of similar submission pairs on technical review assessments.

sations with relatively long code to build. There was also a possibility of having high similarity due to the limited variation in the resulting programs. In Assessments 10 and 13, students were expected to use similar syntax; the only differences were the context or the data.

In Assessment 07, 14, and 15, all submissions were relatively unique. This was expected since the assessments were about case studies or mid-test progress. The assessments were larger than others, with general instructions. This entailed a large room for variation in the solutions. Further, the assessments were completed in groups, and only one student per group needed to submit the work.

3.6. *Student Performance (RQ3)*

Students generally performed well during the course, achieving the learning objectives. As shown in Fig. 9, the average marking score for the technical homework assessments was 84, while that for technical review assessments was 85. Our course design seemed

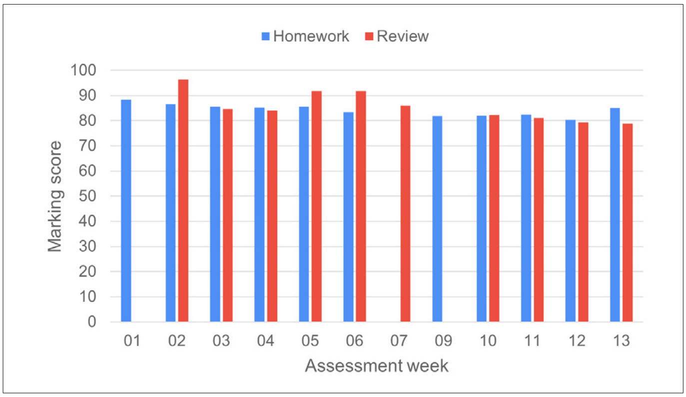

Fig. 9. Student weekly performance on technical review assessments.

helpful for non-computing students to learn data science. GenAI could be useful for supporting student learning, especially in technical areas. This was also supported by the average marking score for the mid-test and the final test. Both were relatively high: 83 for the mid-test and 82 for the final test.

Student performance on the technical homework assessments was comparable to that of the technical review assessments. In a few cases, the performance was even better on the review assessments. Students could maintain their performance even without GenAI, showing limited dependence.

3.7. *Discussion*

Based on our observation, our course design might help students understand GenAI literacy (RQ1). Students knew and understood GenAI usability, reliability, privacy and ethics. Their Likert scale self-assessment was relatively positive (around '4' of '5'). Their understanding of ethics in the era of GenAI was also sufficient. They could correctly respond to 65% of the ethics questions. We, however, acknowledged that the understanding might be affected by other factors such as the clarity of the instructors' guidance, peer discussions, and students' prior experience.

Our course design encouraged students to acknowledge the use of GenAI, and 73% did so. However, the majority of the reported acknowledgements were too short. They could be more comprehensive, addressing the help of GenAI in a more detailed manner. Students acknowledged that GenAI helped with their code-writing process.

Our course design, including its academic misconduct deterrence, also discouraged students from engaging in plagiarism and the misuse of GenAI. The number of students engaged in such misconduct was relatively low. There were more GenAI misuse cases than plagiarism cases, since GenAI was introduced to education only recently.

Many students did not directly copy and paste GenAI results. They aligned the results to their own needs and styles. When observed with a similarity detector, no more than 5% of the submission pairs shared at least half of the content. This might indicate that students used GenAI responsibly and understood the GenAI results, in addition to their preventive reactions to academic misconduct deterrence. Larger assessments with general instructions might be preferred to promote such a benefit further.

Students achieved the course's learning objective, with an average marking score above 80. Although they had no computing and programming backgrounds, they could master the learning materials well with the help of GenAI. The use of GenAI might be encouraged to broaden participation in computing courses for non-computing students.

The course design helps students to know, to understand, to use, and to evaluate GenAI, fulfilling all aspects in AI literacy defined by Long and Magerko (2020). It also helps students to improve their awareness of GenAI, which is expected to change their behaviour as a result of having skills in using GenAI responsibly. Our design fulfils all three dimensions of AI literacy defined by Ng *et al.* (2021).

4. Conclusion and Future Work

We present a GenAI-assisted data science course to promote GenAI literacy for non-computing students. Unique to our course design, it tailors student experiences in GenAI usability, reliability, ethics, and privacy. Students are encouraged to use GenAI while acknowledging it. They are also expected to align GenAI results to their own needs or style. The course design fulfils all AI literacy aspects (Long and Magerko, 2020) and dimensions (Ng *et al.*, 2021).

Based on the course design, students appear to be more familiar with the characteristics of GenAI regarding its usability, reliability, privacy and ethics. Many students acknowledge their use of GenAI, though it is overly general. GenAI is often used to assist the code-writing process, given that the students have limited computing and programming backgrounds. Although GenAI is sometimes associated with plagiarism and misuse, the number of such instances is limited. Students do not directly copy and paste GenAI results. They align the results based on their need and styles. Students achieved the course learning objectives by having an average score of 80.

Our study has a number of limitations which can be addressed in the future. First, the students involved in our study are from the Management undergraduate program. Replicating the study on students from different fields can strengthen the findings. Their prior knowledge and skill set might not be comparable. Based on findings from many replication studies, a framework for integrating GenAI into courses can be developed. Second, our observation is based on a one-semester course offering. Reconducting the same study on the successive offerings might enrich the findings. It is also possible to do quasi-experiments or longitudinal studies to validate the impact of the course design. Third, while our evaluation metrics are satisfactory, we acknowledge that other metrics and statistical analyses exist. These include latent profile analysis. Employing those might help us better understand the impact, including the correlation among the metrics. Fourth, we acknowledge that factors other than the course design might affect the findings. Further, the interpretations may vary. Future work can consider these factors and interpretations during the analyses.

Acknowledgements

The Institute of Research and Community Service (LPPM) at Maranatha Christian University Indonesia supported the research.

References

Almatrafi, O., Johri, A., Lee, H. (2024). A systematic review of ai literacy conceptualization, constructs, and implementation and assessment efforts (2019–2023). *Computers and Education Open*, 100173.

Chiu, T.K. (2024). Future research recommendations for transforming higher education with generative AI. *Computers and Education: Artificial Intelligence*, 6, 100197.

Cronbach, L.J. (1951). Coefficient alpha and the internal structure of tests. *psychometrika*, 16(3), 297–334.
Cubillos, C., Mellado, R., Cabrera-Paniagua, D., Urra, E. (2025). Generative Artificial Intelligence in Computer Programming: Does it enhance learning, motivation, and the learning environment? *IEEE Access*.
Güner, H., Er, E. (2025). AI in the classroom: Exploring students' interaction with ChatGPT in programming learning. *Education and Information Technologies*, 1–27.
Hung, J., Chen, J. (2023). The benefits, risks and regulation of using ChatGPT in Chinese academia: A content analysis. *Social Sciences*, 12(7).
Ji, J., Li, J., Yan, S., Zhang, B., Tian, Q. (2012). Super-Bit locality-sensitive hashing. In: *25th International Conference on Neural Information Processing Systems*. Curran Associates Inc., pp. 108–116.
Jovanovic, M., Campbell, M. (2022). Generative artificial intelligence: Trends and prospects. *Computer*, 55(10), 107–112.
Jury, B., Lorusso, A., Leinonen, J., Denny, P., Luxton-Reilly, A. (2024). Evaluating LLM-generated worked examples in an introductory programming course. In: *Proceedings of the 26th Australasian Computing Education Conference*, pp. 77–86.
Karnalim, O. (2023). Maintaining academic integrity in programming: Locality-sensitive hashing and recommendations. *Education Sciences*, 13(1), 54–15423.
Karnalim, O., Simon, Chivers, W. (2024a). Reporting less coincidental similarity to educate students about programming plagiarism and collusion. *Computer Science Education*, 34(3), 442–472.
Karnalim, O., Toba, H., Johan, M.C. (2024b). Detecting AI assisted submissions in introductory programming via code anomaly. *Education and Information Technologies*, 26(13), 16841–16866.
Korte, S.-M., Cheung, W.M.-Y., Maasilta, M., Kong, S.-C., Keskitalo, P., Wang, L., Lau, C.M., Lee, J.C.K., Gu, M.M. (2024). Enhancing artificial intelligence literacy through cross-cultural online workshops. *Computers and Education Open*, 6, 100164.
Liu, X., Golen, E., Raj, R.K., Fluet, K. (2023). Offering data science coursework to non-computing majors. In: *Proceedings of the 2nd International Workshop on Data Systems Education: Bridging Education Practice with Education Research*, pp. 44–49.
Llerena-Izquierdo, J., Mendez-Reyes, J., Ayala-Carabajo, R., Andrade-Martinez, C. (2024). Innovations in Introductory Programming Education: The Role of AI with Google Colab and Gemini. *Education Sciences*, 14(12), 1330.
Long, D., Magerko, B. (2020). What is AI literacy? Competencies and design considerations. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–16.
Luo, J. (2024). A critical review of GenAI policies in higher education assessment: A call to reconsider the "originality" of students' work. *Assessment & Evaluation in Higher Education*, 1–14.
Manoharan, S., Speidel, U. (2020). Contract cheating in computer science: A case study. In: *2020 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, pp. 91–98. IEEE.
Momen, A., Ebrahimi, M., Hassan, A.M. (2022). Importance and implications of theory of bloom's taxonomy in different fields of education. In: *International Conference on Emerging Technologies and Intelligent Systems*, pp. 515–525. Springer.
Ng, D.T.K., Leung, J.K.L., Chu, S.K.W., Qiao, M.S. (2021). Conceptualizing AI literacy: An exploratory review. *Computers and Education: Artificial Intelligence*, 2, 100041.
Ngo, T.T.A. (2023). The perception by university students of the use of ChatGPT in education. *International Journal of Emerging Technologies in Learning (Online)*, 18(17), 4.
Niloy, A.C., Bari, M.A., Sultana, J., Chowdhury, R., Raisa, F.M., Islam, A., Mahmud, S., Jahan, I., Sarkar, M., Akter, S., *et al.* (2024). Why do students use ChatGPT? Answering through a triangulation approach. *Computers and Education: Artificial Intelligence*, 6, 100208.
Pang, A., Vahid, F. (2024). ChatGPT and Cheat Detection in CS1 Using a Program Autograding System. In: *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*. ACM, pp. 367–373.
Ross, A., Willson, V.L. (2017). One-way ANOVA. In: *Basic and Advanced Statistical Tests: Writing Results Sections and Creating Tables and Figures*. Springer, pp. 21–24.
Shi, X., Liu, J., Liu, Y., Cheng, Q., Lu, W. (2025). Know where to go: Make LLM a relevant, responsible, and trustworthy searchers. *Decision Support Systems*, 188, 114354.
Simon, Cook, B., Sheard, J., Carbone, A., Johnson, C. (2014). Academic integrity perceptions regarding computing assessments and essays. In: *10th Annual Conference on International Computing Education Research*, pp. 107–114. 9781450327558.
Simon, Karnalim, O., Sheard, J., Dema, I., Karkare, A., Leinonen, J., Liut, M., McCauley, R. (2020). Choosing code segments to exclude from code similarity detection. In: *Working Group Reports on Innovation and Technology in Computer Science Education*, pp. 1–19. 9781450382939.

Spatharioti, S.E., Rothschild, D.M., Goldstein, D.G., Hofman, J.M. (2023). Comparing traditional and llm-based search for consumer choice: A randomized experiment. *arXiv preprint arXiv:2307.03744*.

Tzirides, A.O.O., Zapata, G., Kastania, N.P., Saini, A.K., Castro, V., Ismael, S.A., You, Y.-l., dos Santos, T.A., Searsmith, D., O'Brien, C., *et al.*(2024). Combining human and artificial intelligence for enhanced AI literacy in higher education. *Computers and Education Open*, 6, 100184.

Ullman, J.B., Bentler, P.M. (2012). Structural equation modeling. *Handbook of Psychology, second edition*, 2.

Yusuf, A., Pervin, N., Román-González, M., Noor, N.M. (2024). Generative AI in education and research: A systematic mapping review. *Review of Education*, 12(2), 3489.

Zhai, C., Wibowo, S., Li, L.D. (2024). The effects of over-reliance on AI dialogue systems on students' cognitive abilities: a systematic review. *Smart Learning Environments*, 11(1), 28.

O. Karnalim is an informatics engineering lecturer at Maranatha Christian University, Indonesia. He is also the dean of the Faculty of Smart Technology and Engineering. His primary research interests are computing education and programming, and he has published more than 100 papers on these topics. He is the editor-in-chief of Sage's Journal of Educational Technology Systems.

H. Toba is a machine learning professor at Maranatha Christian University, Indonesia. His primary research interests include GenAI-based learning technologies and machine learning.

Peter is a management lecturer at Maranatha Christian University, Indonesia. His primary research interests include financial management and technology.

J.H. Widjaja is a management lecturer at Maranatha Christian University, Indonesia. His primary research interests include strategic management.

T.M. Kuang is an accounting lecturer at Maranatha Christian University, Indonesia. His primary research interests include behavioural accounting, accounting education, and technology-mediated business education.

D.T. Liputra is an industrial engineering lecturer at Maranatha Christian University, Indonesia. His primary research interests include supply chain management and technology-mediated education.

V. Arisandhy is an industrial engineering lecturer at Maranatha Christian University, Indonesia. His primary research interests include operations research and technology-mediated education.