Informatics Insight Journal (IIJ)

Vol. xx (xx) yyyy, 1–10

Received: d month yyyy | Revised: d month yyyy | Accepted: d month yyyy | Published online: d month yyyy

Article Title

DOI: 10.2356/IIJ.EXXXX

Author1 Name, Author2 Name and Author3 Name*

1 Department, Name of organization, Country, Email address.
2 Department, Name of organization, Country, Email address.
3 Department, Name of organization, Country, Email address.

***Corresponding author:** Hariyadi Fajar Nugroho, Department of Science, University of Muhammadiyah, Indonesia. Email: hfn962@gmail.com

Abstract: This study investigates the effectiveness of adaptive learning systems powered by artificial intelligence (AI) in enhancing the programming skills of undergraduate informatics students. Conducted at a major Indonesian university, the research involved 60 participants enrolled in an introductory programming course. Students were divided into two groups: one used a conventional e-learning platform, while the other utilized an AI-based adaptive learning system that adjusted difficulty and feedback in real time. Data were collected through pre-tests, post-tests, and learner engagement analytics. Results indicate that the adaptive system significantly improved both student performance and engagement compared to the traditional platform ($p < 0.05$). Furthermore, qualitative feedback revealed that students found the personalized feedback and pacing to be highly beneficial. The findings suggest that intelligent learning environments can play a critical role in strengthening informatics education

Keywords: solid waste management, recycle, in-service teachers, belief, concern, practice

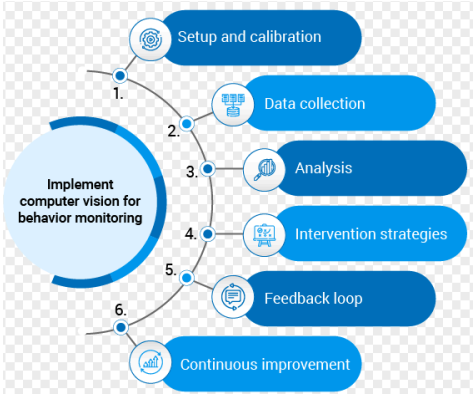
1. Introduction

Programming is a fundamental skill in computer science and informatics education. However, traditional teaching methods often fail to accommodate individual learning differences [1]. To address this, educators are turning to adaptive learning systems that utilize artificial intelligence to personalize learning experiences. This study explores how an AI-based adaptive learning platform can impact students' programming competency and engagement in a real classroom setting.

2. Literature Review

Research has shown the potential of AI in transforming education through personalized learning [2]. Adaptive systems dynamically tailor content and feedback based on student progress, thereby improving outcomes. Kumar and Singh [3] reported that such systems increase learner retention and satisfaction in programming education. Despite this, there remains a gap in research within Southeast Asian contexts [4].

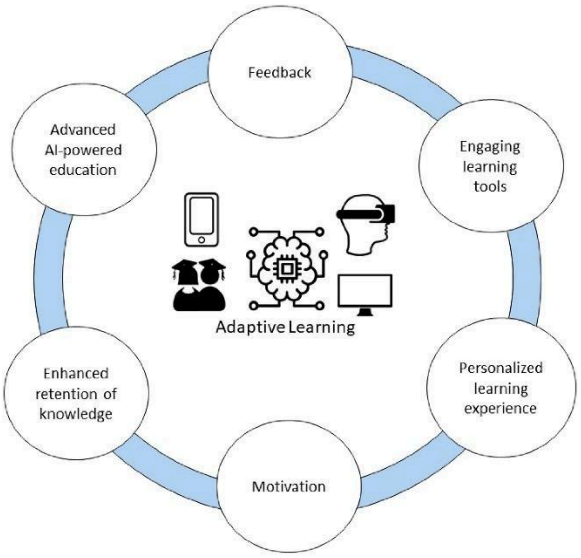
Figure 1
Example



2.1. Theoretical framework

This study employs a quasi-experimental design using mixed methods. A pre-test and post-test control group design was used to assess differences between students using a traditional e-learning platform and those using the AI-based system. Quantitative data were collected via test scores and engagement surveys, while qualitative insights were drawn from open-ended questionnaires.

Figure 2
Example



3. Research Methodology

3.1. Research design

Sixty students were divided randomly into two groups. Group A used a standard Learning Management System (LMS), while Group B used an AI-based adaptive learning platform that included real-time feedback, content adjustment, and analytics-based progress tracking [2]..

3.2. Participants

Participants consisted of 60 undergraduate informatics students at [University Name], taking an introductory programming course. They were randomly assigned into two groups of 30 students each. None had prior experience with adaptive learning systems..

3.2.1. Instruments

In-service first year postgraduate science teacher's attitude towards SWMR were adapted from the Scale for the Attitudes of Pre-service Teachers towards Solid Wastes and Recycle developed by Karatekin (2013). The scale consists of 30 questions and 3 themes. The theme belief consists of 7 items, the theme concern consists of 10 items, and the theme practice has 13 items. The instrument was self-reported with structured five-point Likert scale ranging from Strongly Disagree, Disagree, Neutral, Agree to
Strongly Agree. The values assigned to the options were 1, 2, 3, 4, and 5 respectively. The Cronbach's Alpha for instrument reliability was recorded at 0.87 for the 30 items questionnaire.

Table 1
Interpretation of the mean scale for belief, concern, and practice

Instrument	Description	Type	Reliability
Pre-test	25 multiple-choice questions on programming fundamentals	Quantitative	$\alpha = 0.83$
Post-test	Matched 25 questions of similar difficulty	Quantitative	$\alpha = 0.86$
Learner Engagement Survey	15 Likert-scale items on motivation and interaction	Quantitative	$\alpha = 0.79$
Open-ended Questionnaire	3 reflective questions on user experience	Qualitative	-

4. Conclusion

The results indicate that AI-based adaptive learning systems can significantly enhance programming education. Students using the adaptive system scored higher in the post-test and demonstrated increased engagement. These findings are consistent with prior research that emphasizes the effectiveness of intelligent tutoring systems [3], [4]..

Recommendations

Academic institutions should explore the integration of AI-based adaptive technologies in informatics curricula. Further research could include longitudinal studies and examine their applicability in advanced programming, data structures, or machine learning courses.

Acknowledgement

The authors thank the Informatics Department of [University Name] for supporting this study, as well as the students who volunteered to participate..

Funding Support

This research was funded by [Funding Agency], under Grant No. [XXXX]..

Ethical Statement

This study was approved by the Institutional Ethics Committee. Informed consent was obtained from all participants prior to data collection.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The datasets generated during this study are available from the corresponding author upon reasonable request.

References

[1] Nguyen, T., Huynh, Q., & Pham, L. (2020). AI-assisted Teaching Models in Southeast Asia. *International Journal of Informatics Research*, 9(3), 201–215.

[2] Chen, J., Liu, H., & Zhang, L. (2021). Artificial Intelligence in Education: Adaptive Systems for Personalized Learning. *Journal of Educational Technology*, 18(2), 123–137.

[3] Kumar, R., & Singh, A. (2019). Adaptive Learning Platforms in Programming Education: A Review. *Computing Education Review*, 14(1), 45–60.

[4] Susilo, D., & Ramadhani, N. (2021). Implementation of Adaptive Systems in Indonesian Higher Education. *Journal of Informatics Development*, 12(4), 234–240.

[5] Vygotsky, L. S. (1978). *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.

[6] Piaget, J. (1973). *To Understand is to Invent: The Future of Education*. Penguin Books.

The format should follow the manuscript tempale that contains:

TITLE: The title of the article should contain as few words as feasible while still accurately summarizing its substance (Cambria, Center, Bold, 16pt)

AUTHORS: The author's names should be accompanied by the author's institutions, institutions address, and email addresses, without any academic titles and job titles.

ABSTRACT: It must be able to stand alone since abstracts are frequently given apart from articles. A well-written abstract allows the reader to identify the main points of a document, assess its relevance to their interests, and decide whether to study the text in its full quickly and properly. The abstract should clearly identify the aim, the suggested method or solution, and highlight the most important findings and conclusions. It should be informative and entirely self-explanatory. The Abstract should be between 100 and 250 words. Avoid using references, but if you must, list the author(s) and the year (s). Standard nomenclature should be used, and non-standard or uncommon abbreviations should be avoided, but if essential they must be defined at their first mention in the abstract itself. No literature should be cited.

KEYWORDS: The keyword list provides the opportunity to add 5 to 7 keywords, used by the indexing and abstracting services, in addition to those already present in the title.

INTRODUCTION: The introduction should include at least 3 paragraphs containing: (1) Paragraph I contain the background/problems/facts that support the research/study; (2) Paragraph II contains the state of the art/related works which contains at least 5 related previous studies; (3) Paragraph III contains the research position/gap/differentiation and defines the purpose of the research/study.

METHOD: The method explain the research chronological, including research design, research procedure (in the form of algorithms, pseudocode or other). Then, describe the case study and data sources (data collections) used and how to test or acquire data.

RESULT AND DISCUSSION: Result and discussion section explain the complete results of research and at the same time is given the comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily. Please, write this section systematically according to the method used and in the discussion there must be references cited as a comparison/analysis of the linkages of the findings with previous research.

CONCLUSION: Conclusion provide a statement that what is expected, as stated in the "Introduction" chapter can ultimately result in "Results and Discussion" chapter, so there is compatibility. Do not repeat the abstract. Conclusion should contains results and discussion (findings), contributions, implications, can also explain the "lack of study" of the research. Then, there should be suggestions for further research.

REFERENCES: References should: (1) Minimum 20 references; (ii) 80% are sourced from reputable journal articles (clear sources and DOI) up to date, maximum in the last 5 years; (iii) 20% comes from sources other than journal articles (books, news portals, etc.); and (iv) Using the Mendeley reference manager.

The table should be numbered, for example, Table 1. Category. The description of the table (number and title of the table) must be placed in above of the table.

Images should be numbered, for example, Figure 1. Context Diagram. The description of the image (the number and the title of the image) must be placed in below of the image.

Mathematical formulas must be written clearly, number and information must be described in the formula.

Citation and reference writing are written using IEEE Style with numbering. The numbering refers to the list of references used. The references are written by using numbers with square brackets, for example: [1] First reference.