Jurnal
Guru Komputer
Journal homepage: https://ejournal.upi.edu/index.php/JGrKom

Development of Web-Based Academic Information System Using Django and R&D for Student Achievement Records with Integrated Supervision Module

Latifahny Aridia Alfitri[*], *Zahra Khaerunnisa*

Universitas Pendidikan Indonesia, Indonesia
*Correspondence: E-mail: latifahny_aridia@upi.edu

ABSTRACT

Student achievement data collection is a crucial component of academic management, particularly at the Study Program level. However, many institutions still rely on the manual and fragmented processes, leading to inefficiencies, data loss, and lack of structured supervision. This study aimed to develop a web-based internal information system called SIMP PILKOM (*Sistem Informasi Mahasiswa berPrestasi Pendidikan Ilmu Komputer*) to support the documentation of student achievements and integrated supervision activities. The system was developed using a Research and Development (R&D) approach with the ADDIE model; consisting of analysis, design, development, implementation, and evaluation phases. The system was built with the Django framework, supported by SQLite and MySQL databases, and implemented with HTML and CSS interfaces. Key features include competition data entry by students, supervisor selection, and documentation of supervision sessions. To evaluate the system's effectiveness, qualitative feedback was gathered through structured interviews with both students and lecturers, focusing on user satisfaction, usability, relevance to academic workflows, and potential for system adoption. The result indicated high levels of user satisfaction and confirmed that SIMP PILKOM successfully reduced administrative workload, improved tracking of supervision activities, and enhanced interaction between lecturers and students. The system lays a strong foundation for broader integration in academic information system at the institutional level.

ARTICLE INFO

Article History:
Submitted/Received 11 Jun 2025
First Revised 19 Jun 2025
Accepted 20 Jun 2025
First Available Online 29 Jun 2025
Publication Date 01 Jul 2025

Keyword:
ADDIE,
Achievement Data Collection,
Django,
Information System.

1. INTRODUCTION

Student achievements and publications are among the key components for a Study Program in the higher education environment. One of the main challenges in documenting these accomplishments lies in the suboptimal system currently used for managing student achievement and publication data. The data collection process is often conducted manually, relying on physical documents or dispersed digital files, making it vulnerable to data loss and information redundancy. This situation hampers the Study Program's ability to track student achievements accurately and consistently, causes delays in decision-making, and complicates the documentation process for purposes such as accreditation or the provision of awards to outstanding students.

The absence of an integrated data management system can also compromise transparency and accountability in overseeing academic achievements. Furthermore, it may diminish student motivation, as there is a lack of formal recognition or appreciation for their accomplishments. Acknowledging student achievements not only benefits the individual but also enhances the reputation of both the Study Program and the institution at large. In the framework of quality management in higher education, an effective information system is essential to support the efficient collection, storage, processing, and analysis of data.

In response to technological advancements, the development of internal web-based information systems has emerged as a promising solution. Such systems provide centralized, real-time, and flexible data access for students, lecturers, and academic administrators alike. Utilizing web development frameworks like Django, which is based on Python, enables the creation of applications that are fast, secure, and maintainable; thanks in part to its built-in ORM, CSRF/XSS protections, and modular design, as demonstrated by Duisebekova, Khabirov, and Zholzhan (2021). Additionally, the Research and Development (R&D) approach is considered particularly appropriate for this type of system because it supports systematic planning, iterative development, and active user involvement; essential elements when creating solutions tailored to real academic workflows (Umar et al., 2023; Abuhassna & Alnawajha, 2023). This method enables phase-by-phase refinement and validation, ensuring that the system meets user needs and institutional requirements effectively.

This study aims to develop a web-based internal information system for the Study Program that supports the integrated documentation of student achievements and publications. The system is designed not only for internal administrative use but is also directly accessible by students and lecturers. Studiyanti et al. (2019) highlight how usability testing directly contributes to satisfying user experiences. It includes a supervision module that enables active lecturer involvement in mentoring students participating in competitions. Key features of the system allows students to input details of the competitions they are participating in and select a lecturer to serve as their supervisor. Lecturers, in turn, can view a list of students currently participating in competitions and requiring supervision. Both students and lecturers are able to document the supervision activities that take place throughout the competition process. The resulting system is expected to offer an innovative solution to the data management challenges faced by the Study Program level, particularly in terms of student achievement tracking, while also contributing to the strengthening of academic governance in higher education.

2. METHODS

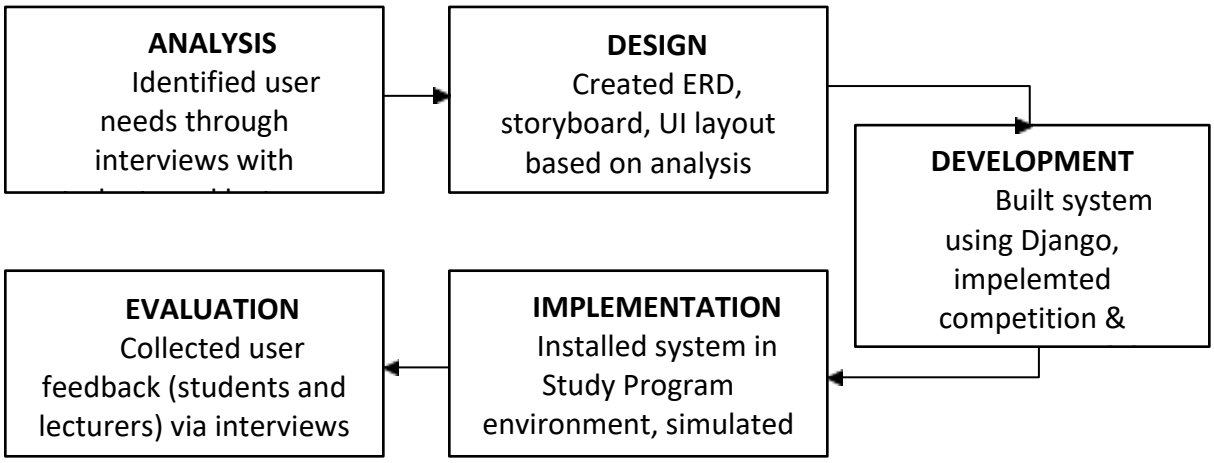


Figure 1. Flow diagram of the system development process based on the ADDIE model.

This study adopts a Research and Development (R&D) approach using the ADDIE model, consisted of Analysis, Design, Development, Implementation, and Evaluation, as the framework for developing the information system. The ADDIE model is considered suitable for this study because it supports a phased development process while continuously addressing user needs. A similar approach was employed by Laipaka (2017) in developing a web-based e-learning system, demonstrating the model's effectiveness in designing structured and systematic educational information systems. The model's relevance is further reinforced by the findings of Akromudin and Munggaran (2024), whose research showed that implementing the ADDIE model significantly improved the effectiveness and efficiency of asset management in specialized institutions. The resulting web-based system in their study successfully addressed managerial issues in an organized manner and contributed meaningfully to system development tailored to institutional requirements. Meta-analyses have shown that the ADDIE model consistently yields structured yet flexible designs suitable for online learning environments (Spatioti et al., 2022; Feng & Sangsawang, 2023), while Adeoye (2024) emphasizes its benefit in enhancing pedagogical effectiveness.

Branch (2009) noted that the ADDIE model consists of five core stages that are sequential yet iterative, allowing for adjustments based on ongoing evaluations. Gustafson and Branch (2002) emphasized the model's prominence in educational system development due to its structured yet adaptable nature. Furthermore, Molenda (2003) highlighted that although ADDIE was not initially designed as a formal model, its consistent application has established it as a standard framework in instructional design and information system development. The following is an explanation of each stage in the ADDIE model:

a. Analysis

   The initial stage involved conducting discussions and coordination sessions with system developers and primary users, namely lecturers and students of the Computer Science Education (*Pendidikan Ilmu Komputer*, can be abbreviated as PILKOM) Study Program, to identify the needs for a data collection system related to student achievements and publications. These discussions focused on gathering information regarding the types of data required, desired system features, and user interface preferences. An initial

system concept and a preliminary database design were drafted as foundational references for the system development process.

b. Design

Following the needs analysis, the system was designed by detailing its data structure, interface layout, and the necessary hardware and software architecture. An Entity Relationship Diagram (ERD) was constructed to model the database, accompanied by a storyboard that illustrated the interface and flow of system usage. The resulting design documentation was reviewed and validated collaboratively to ensure alignment with user expectations and development goals.

c. Development

At this stage, system development was carried out using the Django framework, which is based on the Python programming language, in accordance with the validated design specifications. The database structure was implemented as outlined in the ERD, and the key system features were developed according to the storyboard. Internal testing was conducted during development to ensure that system functionalities operated correctly and met the defined requirements. Continuous collaboration between developers and potential users was maintained to refine and enhance the system based on real-time feedback.

d. Implementation

The developed system was deployed on hardware infrastructure available within the Study Program environment. A data migration process was conducted by inputting previously documented information into the new system following the predesigned data flow. This phase also included system orientation for students, lecturers (particularly those serving as competition supervisors to the students) and Study Program administrative staff. To support optimal use, training sessions and user manuals were provided as part of the implementation and system familiarization process.

e. Evaluation

Following system deployment, an evaluation was conducted to assess its effectiveness and efficiency. Feedback was collected from users specifically students and lecturers of the PILKOM Study Program through interviews aimed at measuring user satisfaction and identifying any obstacles or shortcomings in the system. The evaluation outcomes served as the basis for iterative improvements and further development of the information system.

The Django framework was selected as the core foundation due to its efficiency, modular structure, and strong support for developing object-oriented web applications. Its Model-View-Template (MVT) architecture enables a clear separation between data logic, user interface, and business processes, thereby streamlining system development and maintenance. Additionally, the integration of the Django Rest Framework facilitates seamless system interoperability through API development. Tanjung and Wibowo (2025) demonstrated that Django significantly enhances both the efficiency and accuracy of web-based data management systems. Further support for the adaptability and accessibility of academic information systems is evident in the findings of Harahap et al. (2024), which confirm that web-based platforms can effectively address the structured and evolving needs of educational institutions, including those at the Study Program level. Similar systems built with Django have demonstrated high user acceptance and system robustness (Dela Rosa & Abad, 2025).

To evaluate the effectiveness of the developed system, a qualitative evaluation was conducted involving both students and lecturers as end users. The evaluation focused on three key aspects: user satisfaction, system usability, and relevance to academic workflows. Data were collected through structured interviews and observation during system trials. These evaluation components were selected to ensure that the system met the actual needs of its users and integrated well into existing academic activities. Talaohu and Gunawan (2024) emphasized that user-centered evaluations in academic systems are effective in identifying usability issues and increasing overall system acceptance, which supports the approach taken in this study.

3. RESULTS AND DISCUSSION

This study introduces the Computer Science Education Student Achievement Information System (*Sistem Informasi Mahasiswa berPrestasi Pendidikan Ilmu Komputer*, hereinafter abbreviated as SIMP PILKOM), a web-based internal platform designed to manage and document student achievements and academic guidance activities within the Study Program. SIMP PILKOM has been successfully developed as a digital solution to support the documentation of student achievements and publications. This system is designed to meet the needs of its users, students and lecturers, in carrying out various administrative tasks related to achievement recording. Key features include the ability to document competition participation, select supervising lecturers, and record mentoring sessions conducted between supervisors and student participants. The implementation of role-based access ensures that students and lecturers interact with the system according to their specific roles and responsibilities, enhancing both usability and data integrity within the Study Program environment.

Figure 2. Role-based Access Interface.

As seen in **Figure 2**, students are granted the ability to document the competition activities they participate in and to select supervisors whose expertise aligns with the relevant competition field. Lecturers, on the other hand, can access the list of students under their supervision and monitor progress through the recorded mentoring sessions. Technically, SIMP PILKOM was developed using Django as the primary Python-based framework. The system utilizes SQLite and MySQL for database management, while the web interface is designed with HTML and CSS to ensure responsiveness and user-friendliness.

The development of SIMP PILKOM follows the ADDIE model, consisting of five stages: analysis, design, development, implementation, and evaluation. In the analysis phase, in-

depth discussions were conducted with primary users to identify system requirements, challenges, and expectations. The results indicated that the Study Program previously lacked a centralized system for documenting student achievements, resulting in frequent data loss, delayed reporting, and limited lecturer involvement in competition supervision. These findings informed the subsequent design phase, which covered data structures, navigation flow, and user interface elements. Entities and their relationships were visualized through an Entity Relationship Diagram (see **Figure 3**), while a storyboard was developed to illustrate the interface and process flow from the user's perspective. The design was then validated through consultations with both lecturers and students to ensure its alignment with user needs and expectations.

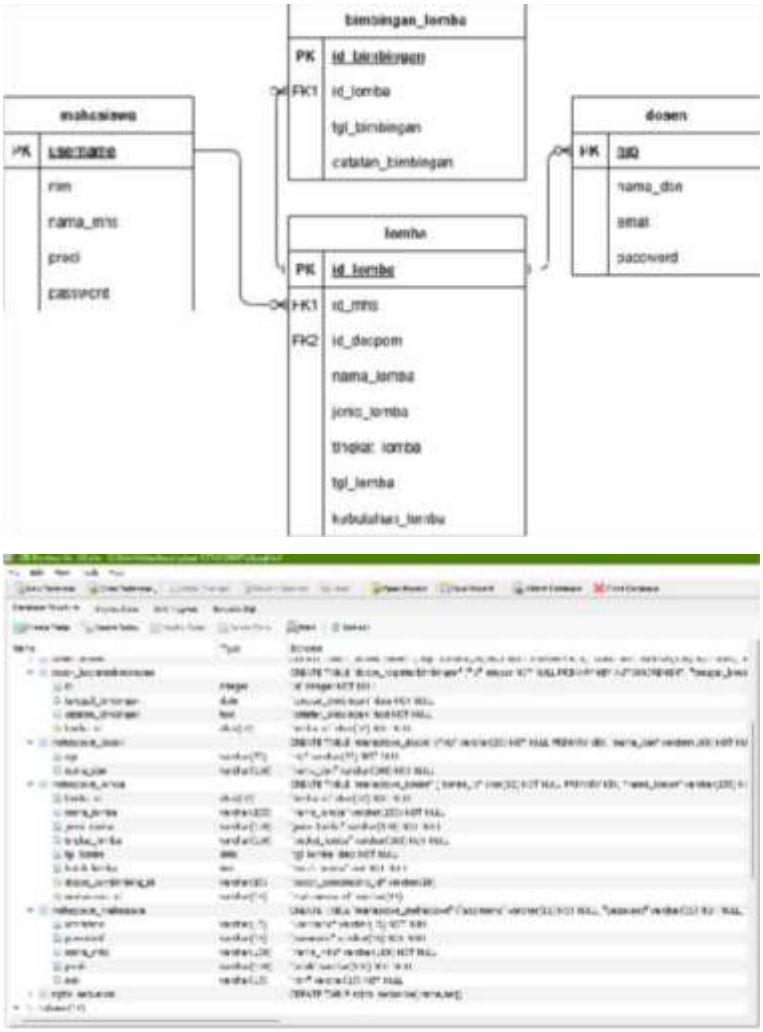


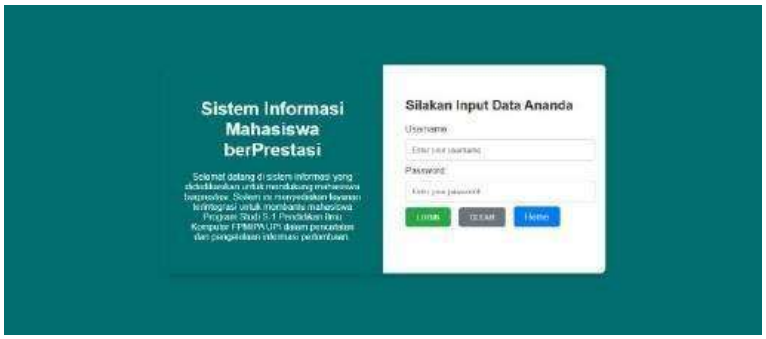
Figure 3. ERD of the Information System (Students, Lecturers, and Competition).

The development of SIMP PILKOM was conducted iteratively using the Django framework. Each feature was implemented based on the validated system design, beginning with the student competition data input module, followed by the supervisor selection feature, and concluding with the guidance session recording functionality. Upon completion of these core features, a series of internal tests were conducted to ensure that all processes operated in accordance with the specified requirements.

During the implementation phase, SIMP PILKOM was installed on devices within the Study Program environment, accompanied by the entry of initial data for simulation purposes. System orientation was provided through brief training sessions and the distribution of user

guides to both students and supervisors. Initial observations indicated that users found SIMP PILKOM to be user-friendly, informative, and well-aligned with the administrative needs of the Study Program. The interface design flow can be seen in **Table 1**.

Table 1. Interface Design Flow.

Design Output	
Interface	**Description**
	The home page interface of SIMP PILKOM provides a role-based login access point, allowing users to enter the system according to their respective roles, either as a Lecturer (*Dosen*) or a Student (*Mahasiswa*). This interface is designed to clearly distinguish user types and streamline access to features relevant to each role. By selecting the appropriate login option, users are directed to system functionalities that align with their responsibilities. Lecturers gain access to supervision and monitoring tools, while students can manage competition records and request guidance accordingly.
	The login page interface for students, where users can enter their registered username and password to access the system.

	The interface of the student dashboard allows users to view a list of competitions they have participated in, with the ability to modify or delete competition entries. It also provides access to the recorded supervision activities with lecturers for each competition.
	The interface for students to add a competition entry, allowing users to input relevant competition details according to the fact and their needs.
	The interface of the lecturer login page allows users to enter their existing username and password to access the system.
	The interface of the lecturer dashboard allows users to view a list of competitions participated in by their supervised students and to record supervision activities related to competition mentoring.
	The interface of the supervision activity history page, displaying records of previously conducted mentoring sessions.

	The interface of the supervision activity form page, designed specifically for lecturers, enables users to record new supervision sessions into the system from the lecturer's perspective.

Evaluation was conducted through interviews with several system users, including those real students and lecturers within the system as respective roles. Usability testing directly contributes to satisfying user experiences, supporting the choice to evaluate SIMP PILKOM through interviews focused on user satisfaction and system adoption. Therefore, a simulation through Zoom meeting was conducted as seen in **Figure 4**. Overall, the results indicate that the system has successfully reduced administrative burdens related to competition documentation, facilitated the tracking of supervision activities, and strengthened communication between lecturers and students in the context of achievement mentoring. Some points for future development include the need for an automatic notification feature and integration with the university's existing academic information systems. The development of these features will be a primary focus in the next version of SIMP PILKOM.

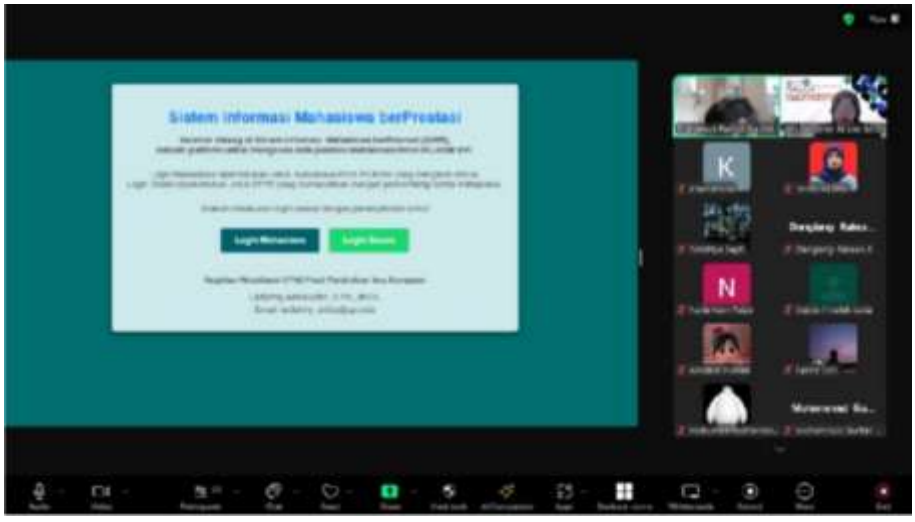


Figure 4. Presentation and Simulation of the Student Achievement Information System (SIMP PILKOM) to Students via Zoom.

Overall, the development and implementation of SIMP PILKOM demonstrates that the system effectively addresses the fundamental needs of the Study Program in documenting student achievements in a structured and well-documented manner. Positive feedback from early users indicates that the system is not only technically beneficial but also enhances the functional interaction between academic supervisors and students in the context of competition mentoring. This system serves as an initial foundation for broader integration of information systems within the academic environment of higher education institutions, while also opening opportunities to improve the quality of academic management through digital technologies that are responsive to the evolving needs of higher education.

4. CONCLUSION

The results of this study indicate that the development of the Computer Science Education Student Achievement Information System (SIMP PILKOM) serves as an effective solution to the challenges of documenting student achievements and publications within the Study Program of Computer Science Education. The system was designed based on actual user needs and developed using the ADDIE model, which includes the stages of analysis, design, development, implementation, and evaluation. SIMP PILKOM enables students to record the competitions they participate in, select a needed lecturer as supervisors aligned with the competition field, and document supervision activities carried out throughout the competition process. Meanwhile, lecturers can systematically monitor student progress through the supervision feature.

This system successfully addresses two core needs: providing a platform for integrated documentation of student achievements and creating a collaborative space between lecturers and students in the competition mentoring process. The evaluation of system usage shows that SIMP PILKOM not only simplifies data entry and reporting but also improves the efficiency of academic administration within the Study Program. This initiative demonstrates that an information technology-based approach, designed through participatory methods, can make a tangible contribution to strengthening academic governance at the program level. Although the system still has certain limitations, such as the absence of an automatic notification feature and integration with the university's academic system, SIMP PILKOM has established a strong foundation for the broader and more sustainable development of academic information systems.

5. ACKNOWLEDGMENT

The author extends sincere gratitude and appreciation to all parties who contributed to the research and development of the system. Special thanks are directed to Prof. Lala Septem Riza, M.T., and Dr. Wahyudin, M.T., as supervisors and Head of Computer Science Education Study Program, for their continuous guidance and support throughout the research process, which made the successful completion of this research and system development possible. The author also wishes to express heartfelt thanks to fellow lecturers, particularly Yogi Prasetyo, M.Kom., and Yuli Sopianti, S.Pd., M.Kom., whose involvement was essential in ensuring the smooth progress of this research. Finally, with deep appreciation, the author thanks the editors and reviewers of Jurnal Guru Komputer for their acceptance and valuable feedback, which helped bring this research to its published form.

6. AUTHORS' NOTE

The authors declared that there were no conflicts of interest related to the publication of this article. Furthermore, the authors confirmed that the manuscript was original and free from any form of plagiarism.

7. REFERENCES

Abuhassna, H., & Alnawajha, S. (2023). Instructional design made easy! Instructional Design Models, Categories, Frameworks, Educational Context, and Recommendations for Future Work. *European Journal of Investigation in Health, Psychology and Education*,

13(4), 715-735.

Adeoye, M. A. (2024). Revolutionizing education: Unleashing the power of the ADDIE model for effective teaching and learning. *Jurnal Pendidikan Indonesia*, 13(1), 202–209.

Akromudin, M., & Munggaran, L. C. (2024). Penerapan Metode ADDIE dalam membangun Sistem Informasi Manajemen Aset Berbasis Web pada Unit Kerja Khusus Pusat Pengembangan Kedokteran Indonesia. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 7(3), 1080–1089.

Branch, R. M. (2009). Instructional Design: The ADDIE Approach. *New York: Springer*.

Dela Rosa, A. P. M., & Abad, R. P. P. (2025). Leveraging Technology for Teaching and Learning: Developing a Django-based quiz application for education. *Journal of Information Systems and Education for Mathematics*, 10(2), 1–15.

Duisebekova, K., Khabirov, R., & Zholzhan, A. (2021). Django as secure web-framework in practice. *Bulletin of Kazakh Academy of Transport and Communications*, 116(1), 275–281.

Feng, J., & Sangsawang, T. (2023). Efficiency and student satisfaction of an information technology approach structured by the ADDIE model: A case in English language teaching. *Turkish Online Journal of Educational Technology*, 22(4), 121–133.

Gustafson, K., & Branch, R. (2002). Survey of Instructional Development Models (4th ed.). *ERIC Clearinghouse on Information & Technology*.

Harahap, G. D. F., Adithia, M. K. P., Sarumaha, D., & Mardiah. (2024). Perancangan Sistem Informasi Akademik Dengan Menggunakan Metode Rapid Application Development (RAD) Berbasis Website. *Jurnal Ilmu Komputer dan Sistem Informasi*, 3(3), 215–223.

Laipaka, R. (2017). Penerapan Teknologi Informasi Pembelajaran E-Learning Menggunakan ADDIE Model. *JUPITER: Jurnal Penelitian Ilmu dan Teknologi Komputer*, 9(1), 1–12.

Molenda, M. (2003). In search of the elusive ADDIE model. *Performance Improvement*, 42(5), 34–36.

Spatioti, A. G., Kazanidis, I., & Pange, J. (2022). A comparative study of the ADDIE instructional design model in distance education. *Information*, 13(9), 402.

Studiyanti, L., Azmi, N., Saraswati, D., & Abror, M. (2019). Usability evaluation and design of a Student Information System prototype to increase student satisfaction: Case study of X University. *Industrial Engineering & Management Systems, 18*(4), 676–684.

Talaohu, M. J., & Gunawan, H. (2024). Evaluasi Usability Sistem Informasi Akademik STMIK IM Menggunakan Metode Heuristic Evaluation. *Infotech: Jurnal Informatika & Teknologi*, *5*(1), 33–42.

Tanjung, M. A. A., & Wibowo, A. P. W. (2025). Rancang Bangun Sistem Informasi Pengukuran Indeks Profesionalitas Aparatur Sipil Negara dengan Django. *Jurnal Informatika Teknologi dan Sains (JINTEKS)*, 2(1), 50–58.

Umar, U., Purwanto, M. B., & Al Firdaus, M. (2023). Research and Development: As The Primary Alternative To Educational Research Design Frameworks. *Journal of English Language and Literature (JELL) STIBA-IEC Jakarta*, 8(01), 73-82.