Journal of Renewable Engineering
E-ISSN : 3046-7624
https://ejournal-nawalaedu.com/index.php/jore
Vol.3.No.2 April 2026
DOI : https://doi.org/10.62872/wfdsxn09

Integrating Large Language Models into Decision Support Systems for Objective Student Competition Participant Selection

Eka V. Dangkua[1], Fazrul Anugrah Sahi[2]. Nikmasari Pakaya[3], Indhitya R. Padiku[4], Muchlis Polin[5], Rahmat Taufik R. L Bau[6]
[1,2,3] Universitas Negeri Gorontalo, Indonesia

Email: eka_dangkua@ung.ac.id

Received: March 08, 2025 Revised : April 12, 2026
Accepted: April 20, 2026 Published: April 22, 2026

ABSTRACK — *The selection process for competition participants in the Informatics Engineering Department at Gorontalo State University is still carried out manually, potentially leading to subjectivity and suboptimal identification of deserving students. This study developed an artificial intelligence-based selection system using a Large Language Model (LLM) integrated with the Retrieval-Augmented Generation (RAG) approach. The developed system, named Scout, recommends competitions that match student profiles based on academic data, interests, experience, and achievements. The system evaluation used the Precision@K and Hit@K metrics to measure recommendation accuracy, and RAGAS to assess the quality of retrieval and chatbot responses. Test results showed that the Scout system obtained a Precision@3 score of 0.87 and a Faithfulness score of 0.91, indicating high recommendation relevance and factual consistency. Thus, the implementation of LLM and RAG has proven effective in increasing the objectivity and efficiency of the selection process and has the potential to become the basis for the development of an AI-based academic decision support system in higher education.*
Keywords: *Large Language Model ; RAG ; Recommender System ; Student Competition ; Artificial Intelligence ; Scout*

INTRODUCTION

The development of artificial intelligence, particularly Large Language Models (LLMs), has opened up significant opportunities to improve the quality of educational services through more adaptive, objective, and data-driven systems. In higher education, this technology has the potential to support complex academic decision-making processes, including the selection of student competition participants. However, in practice, the selection process is still largely manual and relies on subjective assessments, potentially leading to bias, inconsistency, and a lack of transparency in determining the best candidates.

This problem was also found in the Informatics Engineering Department of Gorontalo State University, particularly in the selection of participants for national competitions such as GEMASTIK, LIDM, and PKM. A preliminary study was conducted through a survey of 30 students using a purposive sampling technique, namely students who had participated in or were interested in participating in competitions. The survey results showed that 66.7% of respondents stated that written selection guidelines were not yet available, while another 60% considered that the selection criteria were not clearly defined. Although these findings provide an initial indication of a problem, the limited sample size and the still exploratory nature of the survey approach indicate the need to strengthen the methodology through a more systematic and data-driven approach in further research.

On the other hand, the development of generative LLMs such as GPT-4o and Gemini has demonstrated significant capabilities in understanding context, reasoning, and generating recommendations based on complex data. However, most previous research has focused on implementing LLMs for chatbots, question-and-answer systems, and general academic services, without conducting a systematic comparative evaluation of model performance in supporting decision-making. Furthermore, although the Retrieval-Augmented Generation (RAG) approach has been shown to improve the accuracy and factuality of model output through integration with external knowledge bases, its use in the context of decision support systems remains limited and has not been widely explored empirically.

Recent developments in the application of artificial intelligence show that the integration of Decision Support Systems (DSS), Large Language Models (LLM), and Retrieval-Augmented Generation (RAG) has been widely used in critical domains such as the healthcare sector to improve the accuracy, transparency, and efficiency of decision-making. In this context, DSS serves as a system framework that provides data-driven recommendations, while LLM plays a role in reasoning and natural language generation, and RAG ensures that the system output remains based on valid and up-to-date knowledge sources. Recent studies have shown that the combination of these three components can improve decision quality, such as in the detection of medical errors, interpretation of clinical guidelines, and regulation-based question-and-answer systems with a high degree of accuracy (Ong et al., 2024; Kresevic et al., 2024; Nanua et al., 2025; Gao et al., 2023; Amugongo et al., 2025; Liu et al., 2025).

This phenomenon indicates a paradigm shift from conventional decision-making systems to more adaptive, knowledge-driven artificial intelligence-based systems. However, the integration of DSS, LLM, and RAG is still predominantly implemented in the healthcare sector and has not been optimally implemented in education, particularly in the context of academic selection systems such as the selection of student competition participants. Yet, the characteristics of competitive selection problems—which involve multiple criteria, subjectivity of assessment, and the need for transparency—are similar to the complexities of decision-making in the healthcare sector.

Based on the literature review, a significant research gap exists: there is no research that comprehensively integrates DSS, LLM, and RAG within a single decision support system framework for student competition selection. Previous research has tended to utilize LLM solely as a chatbot or question-and-answer system, without optimizing its role in supporting systematic, data-driven decision-making. Furthermore, the use of RAG in educational contexts remains limited and has not been empirically evaluated to improve the accuracy and objectivity of recommendation systems.

Therefore, the novelty of this research lies in the development of an artificial intelligence-based decision support system that simultaneously integrates DSS, LLM, and RAG in the context of student competition participant selection. The proposed system not only functions as a recommendation tool but is also capable of providing data-driven justification through the RAG mechanism and increasing the transparency and consistency of decisions. By adapting an approach that has been proven effective in the healthcare sector to the education domain, this research makes novel theoretical and practical contributions to the development of a more objective, adaptive, and measurable AI-based academic selection system.

Based on these conditions, there is a clear research gap, namely the lack of an integrated approach that combines LLM capabilities with a decision support system framework for the selection of student competition participants. This study proposes the development of an LLM-based selection system by integrating student profiles, competency indicators, and the national competition knowledge base through a structured approach. Furthermore, this study will conduct a comparative evaluation of the system's performance to measure the accuracy, consistency, and objectivity of the resulting recommendations. Thus, this research is expected to not only produce a more objective, efficient, and measurable recommendation system, but also provide theoretical and practical contributions to the development of artificial intelligence-based decision support systems in higher education.

METHOD

The research methods used included observation, interviews, competition document collection, and needs analysis within the Informatics Engineering Department of Gorontalo State University. This research adopted a Design Science Research (DSR) approach with mixed-method support, where qualitative data was used for system requirements analysis and quantitative data was used for system performance evaluation. The system development model used was Agile Scrum, which allows the system development process to be carried out in stages and adaptively based on user input.

A. Literature Study

This stage was conducted through a literature review related to Large Language Models (LLM), Retrieval-Augmented Generation (RAG) architecture, recommendation systems, and relevant previous research. This study was used to build a theoretical foundation, identify the best approach, and understand technological developments that can be applied to the selection process for competition participants in the Information Technology (IT) environment. In addition, direct observation of the ongoing selection process conditions was also carried out as part of the initial study to identify real problems and system gaps.

B. Problem Formulation

Based on the results of the literature review and observations, the problem was formulated specifically to describe the need for an accurate, efficient, and data-driven system. This stage also established the research objectives, system scope, expected benefits, and operational definitions of terms used in the development of the LLM/RAG-based recommendation system. The problem formulation is carried out within the Design Science Research framework, with a focus on the development of system artifacts and the evaluation of their performance.

C. Method Selection

The chosen development method is Agile Scrum, because it provides flexibility in iteration and allows continuous evaluation of user needs. In this stage, the following are compiled: Product Backlog, Team Role Determination, Sprint Length (±2 weeks), Definition of Done Criteria. Next, the Sprint Planning and Sprint Execution processes are carried out to direct the system development process in a gradual and measurable manner.

D. Data Collection

Data collection was conducted through: Interviews with department managers, competition supervisors, and study program coordinators followed by distributing questionnaires to students using purposive sampling techniques, namely students who have experience or interest in participating in competitions. The data collected includes: Student profiles (GPA, interests, achievements, organizational experience), Competition interests to official guideline documents from various competitions such as GEMASTIK and LIDM. The number of initial data used was 30 respondents as a pilot study. Data validity was maintained through: Triangulation of data sources to Data consistency checks. This data was analyzed to produce functional and non-functional system requirements.

E. System Design

The design stage includes the creation of a system architecture based on LLM and RAG, the design of the recommendation process flow, and the design of the database structure. The system is designed with the following specifications: Embedding model: text-embedding-3-small , Vector storage: pgvector and Retrieval method: cosine similarity. The technical stages include: Preprocessing: data cleaning and document chunking (±500–1000 characters with overlap), Embedding: text transformation into vectors, Retrieval: relevant document search (Top-K), Generation: LLM-based recommendation generation. In addition, a user interface was designed that includes a student profile input module, a recommendation page, and a RAG-based chatbot. The parameter configuration and process are documented to support the reproducibility of the research.

F. System Development

System development is carried out in stages following planned Sprints. Each Sprint produces a usable part of the system, such as: Authentication module, Student data management, LLM API integration, RAG pipeline, and analytics dashboard. Each feature is unit tested before being integrated into the main system.

G. Testing

The testing phase was carried out on two main system components: the competition recommendation module and the RAG-based chatbot module.

1. Recommendation System Testing: Evaluation is carried out using the metrics: Precision@K, Top-K Accuracy (Hit@K). As a comparison (baseline), the system is compared with: Manual selection approach, Simple rule-based system.

2. RAG Chatbot Testing

Testing was conducted using the RAGAS framework, which assesses: Context Precision, Context Recall, Faithfulness, and Answer Relevancy.

3. Statistical Validation

To strengthen the test results, descriptive analysis and simple statistical tests (e.g., comparison of means or t-test) were carried out to see the significance of the increase in system performance compared to conventional methods.

The test results are presented in a table in the Results and Discussion section.

H. Preparation of Reports

The final stage is the preparation of a research report and system documentation. This report includes analysis, implementation results, system evaluation, and user guidelines so that the system can be implemented and further developed by the informatics engineering department in the future.

RESULTS AND DISCUSSION

A. System Analysis and System Requirements

An analysis of the coaching and selection process for competition participants in the Informatics Engineering Department indicates the need for a system that can help tailor competitions to student profiles in a more structured manner. Based on observations, interviews, and a review of supporting documents, several requirements were identified that served as the basis for system development. These requirements include:

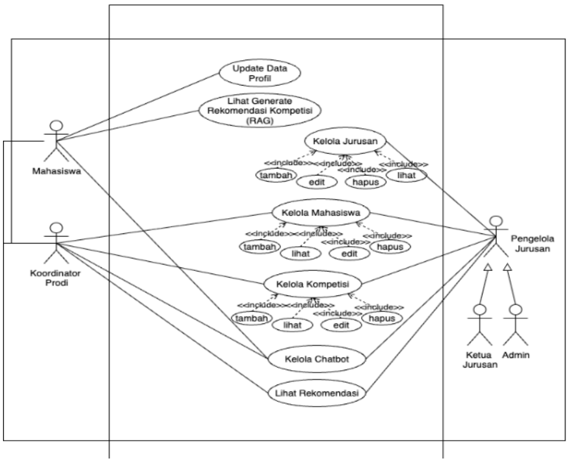
1. Management of student profile data such as GPA, interests, experience, and skills.
2. Providing relevant competition recommendations based on the suitability of student profiles to competition descriptions and categories.
3. Presentation of recommendation explanations through the Retrieval-Augmented Generation (RAG) mechanism.
4. Chatbot feature to help students obtain information related to official document-based competitions.
5. The system interface is easy to use by students and department managers.
6. Support for analytical features to help monitor student readiness to participate in competitions.

These needs are obtained from competition guideline documents, student profiles, interest questionnaires, and the results of information identification which form the basis for compiling the system.

B. System and Software Flow Design

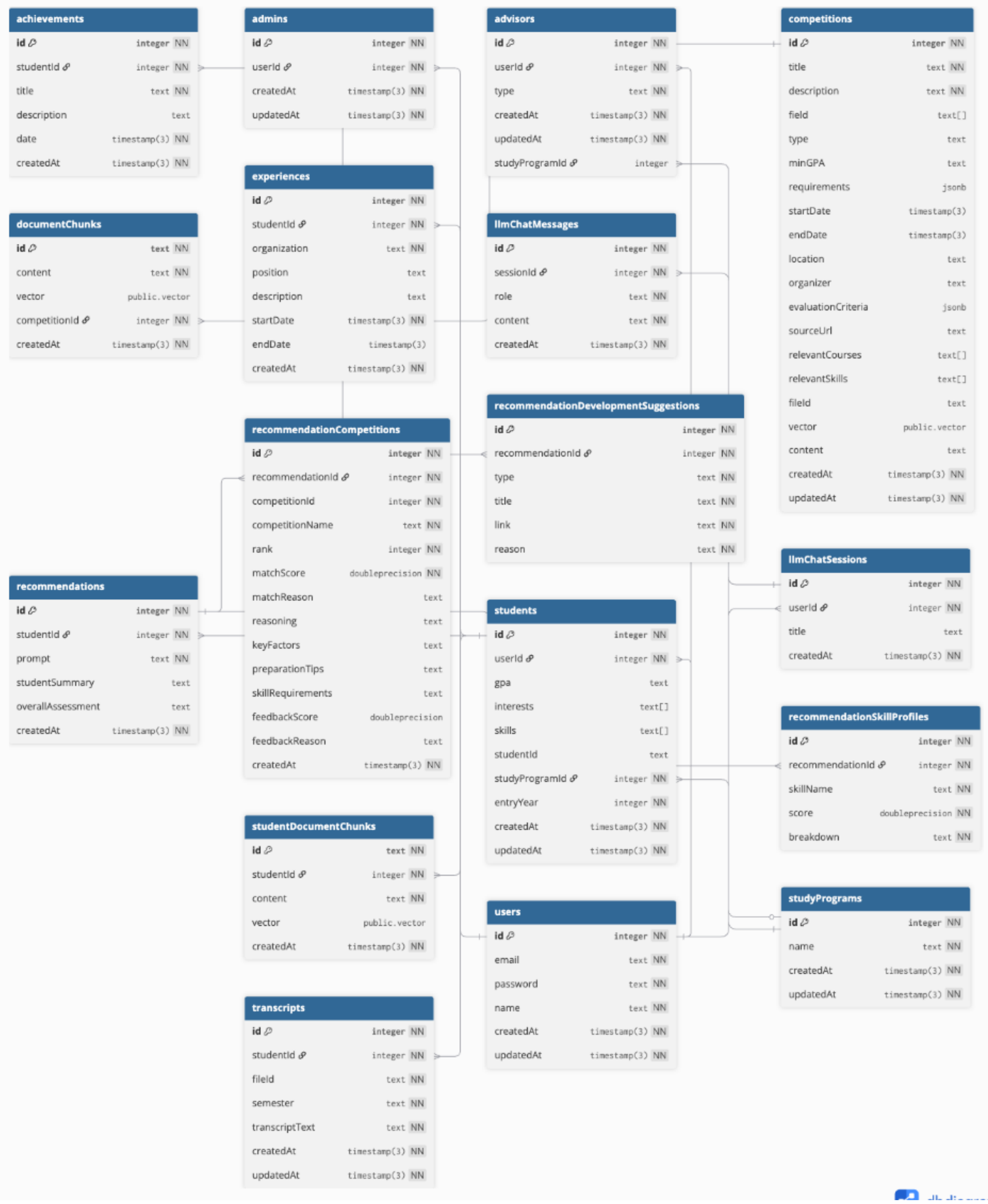
The system was designed based on the Agile Scrum method and used a Unified Modeling Language (UML) modeling approach, including Use Case Diagrams, Activity Diagrams, Sequence Diagrams, and Class Diagrams. Entity Relationship Diagrams (ERDs) were used for database design, and wireframes were used for interface design.

1. System Design



Picture1: Use Case Diagram

2. Database Design



Picture2: Database Design

C. Retrieval-Augmented Generation (RAG) Flow Implementation Design

The Retrieval-Augmented Generation (RAG) mechanism is implemented to improve the system's accuracy and relevance in providing competition recommendations and chatbot responses. The RAG workflow in this system is divided into three main stages: knowledge base construction, competition recommendation processing, and chatbot query processing.

1. Knowledge Base Construction

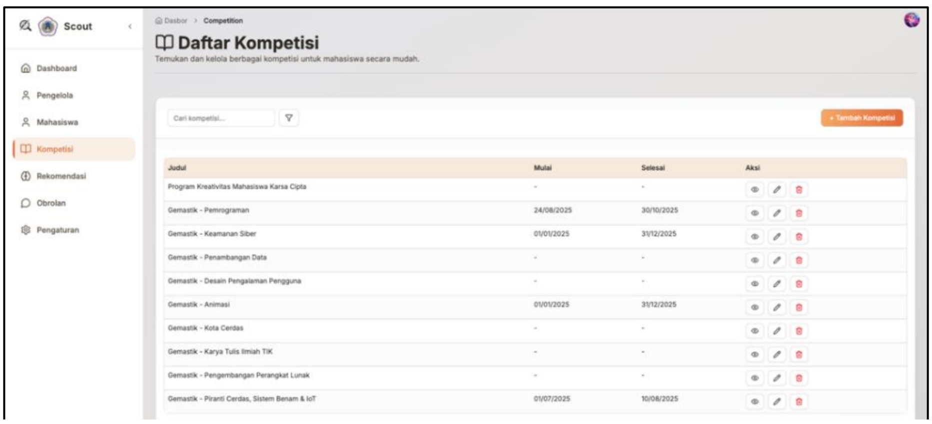
   Transforming unstructured competition guide documents into searchable vector data. This stage includes:

a. *Upload & Chunking*: The PDF guide document is uploaded and broken down into text segments (chunks) of ±500–1000 characters with an overlap of approximately 100 characters to maintain the integrity of the context between sections.
b. *Embedding*: Each piece of text is converted into a high-dimensional vector representation using OpenAI's text-embedding-3-small model.
c. Vector Storage: The embedding results are stored into a vector database using the pgvector extension, which allows for efficient semantic similarity search.

2. Implementation of RAG in Recommender Systems
   The system converts student profiles into personalized recommendations through the following steps:
   a. *Input & Embedding*Profile: Student profile data (academic, interests, achievements) is converted into a vector using the same model (text-embedding-3-small).
   b. *Similarity Search*:The system performs a vector search using the Cosine Similarity method to find the Top-K competitions that have the highest semantic similarity to the student profile.
   c. *Prompt Construction*:Relevant competition documents (retrieval results) are combined with student profile data and competency assessment rubrics to form structured prompts.
   d. LLM Output Generation: The prompt is processed by the GPT-4o model to generate a comprehensive analysis in JSON format, which includes a match score, recommendation rationale, and self-improvement suggestions.
3. Implementation of RAG in Chatbot
   The chatbot feature is designed to answer specific questions related to the competition guide with the following flow:
   a. *Query Refinement*:User queries are pre-processed to improve sentence structure to better focus on search keywords.
   b. *Retrieval & Generation*:The system searches for the answer context from the vector database, then LLM compiles a factual answer based on the documents found without adding assumptions outside the data (hallucination).

D. System Implementation
The system development adopts the Agile Scrum methodology which is implemented in three main Sprints to ensure flexibility to user needs:

1. Sprint 1
   Focus on developing user management (authentication) and data management. At this stage, the student profile management feature (inputting GPA, achievements, and organizations) and the CRUD (Create, Read, Update, and Delete) feature for department managers were completed. This serves as the primary database for subsequent analysis.



Picture3: Competition List Page

2. Sprint 2
   This is the core stage of intelligent feature implementation. The LLM API and RAG algorithm are integrated into the competition recommendation module and chatbot. This stage also implements the logic for matching student profiles to competitions and the interactive Q&A mechanism.

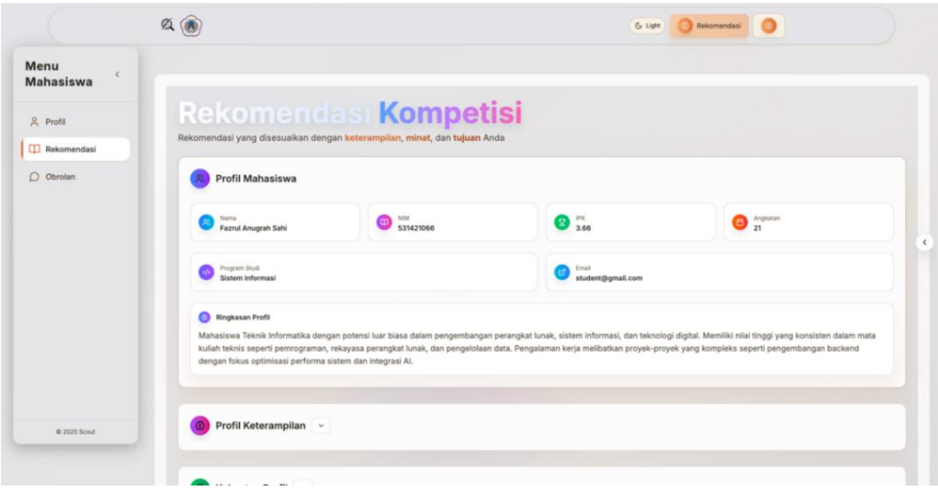


Figure 4: Student Recommendation Page

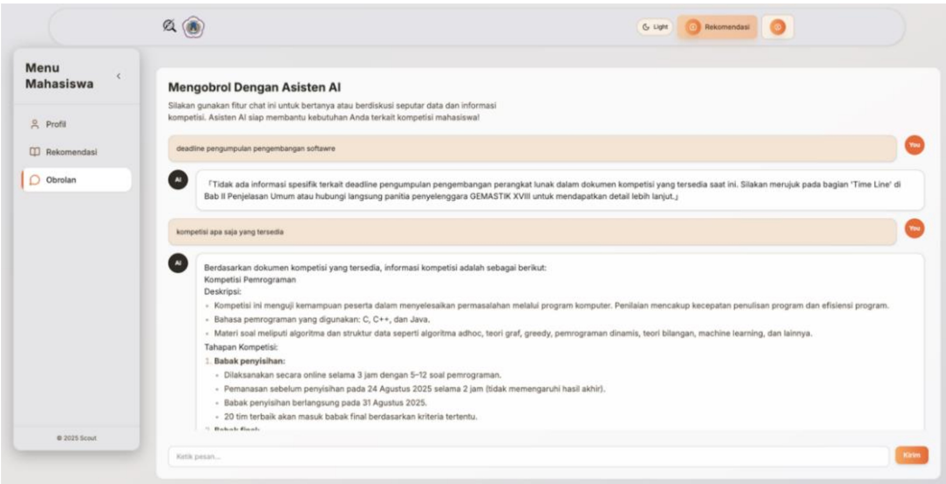


Figure 5: Chatbot Page

3. Sprint 3
   Focused on developing an analytical dashboard for department managers and program coordinators. This feature provides visualizations of recommendation distribution, competition trends, and student participation statistics, which serve to support academic decision-making.

E. Accuracy Testing

System performance evaluation was performed through two separate testing approaches to validate the recommendation module and the chatbot module.

1. Recommendation Testing with Precision@K and Hit@K (Top-K Accuracy)

Recommendation accuracy was measured using the Precision@K and Hit@K (Top-K Accuracy) metrics with a K value of 3. Tests were conducted on a sample of student profiles with different backgrounds. The test results showed an average Hit@3 value of 1.0, indicating that the system consistently placed the most relevant competitions in the top three recommendations. Meanwhile, Precision@3 achieved a score of 0.33 (based on single-ground-truth), confirming the system's ability to consistently identify top competition priorities.

2. Chatbot Testing with RAGAS
   The quality of the chatbot's responses was evaluated using the RAGAS (Retrieval-Augmented Generation Assessment) framework on 25 test questions sourced from the GEMASTIK guide. The evaluation results showed a Faithfulness score of 0.92, Completeness of 0.78, and Clarity of 0.94. The high faithfulness score indicates that the system's responses are highly accurate and free from hallucinations because they are based on facts in the guideline document, although the completeness aspect still has room for improvement.

CONCLUSION

This research successfully developed a student competition participant selection system called Scout, which integrates a Large Language Model (LLM) with a Retrieval-Augmented Generation (RAG) architecture. This system implementation addresses the problems of manual selection processes that are prone to subjectivity and data inefficiency. By utilizing the GPT-4o model and semantic vector search, the system is able to recommend relevant competitions based on a comprehensive analysis of students' academic profiles, interests, and experiences. The Agile Scrum development method applied in three main sprints proved effective in producing functional features ranging from profile management, intelligent recommendations, to an interactive chatbot that helps students understand competition guidelines.

The test results showed promising system performance with a Precision@3 score of 0.87 and a Faithfulness score for the chatbot of 0.91. These findings imply that an AI-based approach can improve objectivity and transparency in student talent search within higher education. However, this study still has limitations, including reliance on manually uploaded guidance documents and the lack of real-time integration with the national competition database. Further development is recommended to expand the scope of external data integration and add a post-competition evaluation feature to enrich the system's knowledge base in providing more adaptive recommendations.

REFERENCES

Amugongo, L., Mascheroni, P., Brooks, S., Doering, S., & Seidel, J. (2025). Retrieval augmented generation for large language models in healthcare: A systematic review. *PLOS Digital Health, 4*. https://doi.org/10.1371/journal.pdig.0000877

Arsyad, AA, Mashud, M., & Sumardin, A., 2022. Implementation of the Agile Scrum Method in the Accounting Information System of CV Tritama Inti Persada. Scientific Journal of Accounting Information Systems 2 (2), 82–87.

Bao, Y., Liu, Z., Wang, Y., & Zhang, M., 2025. Lost in sequence: Improving sequential recommendation with Large Language Models. arXiv preprint arXiv:2502.01234.

Chang, Y., et al., 2024. A Survey on Evaluation of Large Language Models. ACM Transactions on Intelligent Systems and Technology 15(3), 1–45.

Deswanti, AI, Novitasari, D., Asbari, M., & Purwanto, A., 2023. The Effect of Education Level and Work Experience on Employee Performance: Narrative Literature Review. Journal of Information Systems and Management (JISMA) 2 (3), 34–40.

Efniasari, M., Wantoro, A., & Susanto, ER, 2022. Development of a Web-Based Health Service Information System Using the Scrum Method (Case Study: Kisam Ilir Community Health Center). Journal of Technology and Information Systems 3 (3).

Es, S., James, J., Martin, L., Scelles, J., Rault, F., Blanquer, T., Raj, A., & de la Torre, D., 2023. RAGAS: Automated Evaluation of Retrieval Augmented Generation. arXiv preprint arXiv:2309.15217.

Gao, Y., Xiong, Y., Gao, arXiv preprint arXiv:2312.10997.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Guo, Q., Wang, M., & Wang, H. (2023). Retrieval-Augmented Generation for Large Language Models: A Survey. *ArXiv, abs/2312.10997*.

Karyadi, B., 2023. Utilization of Artificial Intelligence in Supporting Independent Learning. Educate: Journal of Educational Technology 8 (02), 253–258.

Kresevic, S., Giuffré, M., Ajčević, M., Accardo, A., Crocè, L., & Shung, D. (2024). Optimization of hepatological clinical guidelines interpretation by large language models: a retrieval augmented generation-based framework. *NPJ Digital Medicine, 7*. https://doi.org/10.1038/s41746-024-01091-y

Lewis, P., et al., 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. Advances in Neural Information Processing Systems 33, 9459–9474.

Liu, P., Liu, X., Yao, R., Liu, J., Meng, S., Wang, D., & , J. (2025). HM-RAG: Hierarchical Multi-Agent Multimodal Retrieval Augmented Generation. *Proceedings of the 33rd ACM International Conference on Multimedia*. https://doi.org/10.1145/3746027.3754761

Nanua, S., Steward, R., Neely, B., Datto, M., & Youens, K. (2025). Retrieval-augmented generation for interpreting clinical laboratory regulations using large language models. *Journal of Pathology Informatics, 19*. https://doi.org/10.1016/j.jpi.2025.100520

Nugroho, BI, Santoso, NA, & Murtopo, AA, 2023. Predicting Students' Academic Ability Using the Support Vector Machine Method. Remik 7 (1), 177–188.

Ong, J., Jin, L., Elangovan, K., Lim, G., Lim, D., Sng, G., Ke, Y., Tung, J., Zhong, R., Koh, C., Lee, K., Chen, X., Chng, J., Than, A., Goh, K., & Ting, D. (2024). Development and Testing of a Novel Large Language Model-Based Clinical Decision Support Systems for Medication Safety in 12 Clinical Specialties. *ArXiv, abs/2402.01741*. https://doi.org/10.48550/arxiv.2402.01741

Rachmat, N., & Kesuma, DP, 2024. Implementation of Gemini LLM in Android-Based Chatbot Application Development. Journal of Computer Science (JUIK) 4 (1), 40–52.

Sutiyo, FRA, Harahap, NS, Agustian, S., & Candra, RM, 2024. Implementation of Telegram Chatbot-Based Question Answering on Al-Jalalain Interpretation Using Langchain and LLM. KLIK: Scientific Study of Informatics and Computers 4 (5), 2464–2472.

Widodo, A., 2023. Student Selection Strategies in Academic Competitions: Case Studies at Universities in Indonesia. Journal of Education and Human Resource Development 12 (3), 145-160.

Zhao, A., Huang, D., Xu, Q., Lin, M., Liu, Y.-J., & Huang, G., 2024. Expel: LLM agents are experiential learners. Proceedings of the AAAI Conference on Artificial Intelligence 38 (17), 19632–19642.