International Journal of Computer, Network Security and Information System (IJCONSIST) ISSN: 2686-3480
Vol. 7 No. 2, Maret 2026, pp. 17-24

Event-Driven Conversational Workflow Architecture for Academic Service Automation Using n8n

Bonda Sisephaputra*
Informatics Engineering
Faculty of Engineering, Universitas Negeri Surabaya
Surabaya, Indonesia
bondasisephaputra@unesa.ac.id

Reisa Permatasari
Information System
Faculty of Computer Science, UPN Veteran Jawa Timur
Surabaya, Indonesia
reisa.permatasari.sifo@upnjatim.ac.id

Ronggo Alit
Informatics Engineering
Faculty of Engineering, Universitas Negeri Surabaya
Surabaya, Indonesia
ronggoalit@unesa.ac.id

Azis Suroni
Informatics Engineering
Faculty of Engineering, Universitas Negeri Surabaya
Surabaya, Indonesia
azissuroni@unesa.ac.id

Rahmat Nugroho Saputra
Informatics Engineering
Faculty of Engineering, Universitas Negeri Surabaya
Surabaya, Indonesia
25051204306@mhs.unesa.ac.id

Fathan Orvala
Informatics Engineering
Faculty of Engineering, Universitas Negeri Surabaya
Surabaya, Indonesia
25051204271@mhs.unesa.ac.id

Dhava Gilang Ramadhan
Informatics Engineering
Faculty of Engineering, Universitas Negeri Surabaya
Surabaya, Indonesia
25051204290@mhs.unesa.ac.id

***Abstract*—The digital transformation of higher education institutions has increased the demand for efficient academic service management systems. However, many existing digital academic service platforms primarily focus on providing online interfaces while the underlying administrative processes remain fragmented and manually coordinated. As a result, academic service requests such as document submissions and information inquiries often lack process transparency and structured workflow management. This study proposes an event-driven conversational workflow architecture for automating academic service processes. The proposed architecture integrates a conversational interface through WhatsApp messaging services, a workflow orchestration engine implemented using the n8n automation platform, and a cloud-based database using Supabase PostgreSQL. The system consists of three main workflows: knowledge base generation, information services using a Retrieval-Augmented Generation (RAG) approach, and document request services with multi-level approval processes. The research adopts the Design Science Research methodology to design and evaluate the proposed system architecture. A prototype system was implemented to demonstrate how academic service requests can be initiated through conversational interfaces and processed through event-driven workflow automation. The evaluation results show that the proposed architecture enables automated workflow orchestration, structured service state management, and improved service transparency.**

***Keywords*—academic services; workflow orchestration; conversational interface; event-driven architecture; design science research**

© 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution ShareAlike (CC BY SA) license (https://creativecommons.org/licenses/by-sa/4.0/).

I. INTRODUCTION

Digital transformation has become a critical strategy for improving service delivery in higher education institutions, prompting universities to adopt information systems for academic administrative processes. Despite these advancements, many academic administrative services still rely on semi-manual coordination among staff and departments, resulting in inefficiencies and limited process transparency.

Conversational interfaces have emerged as a promising approach to improving access to digital services. Chatbots integrated with messaging platforms enable users to interact with information systems using natural language, thereby enhancing accessibility and user experience [1]. This

interaction model simplifies traditional system interfaces and allows users to access services more intuitively.

Workflow management systems have been widely applied in organizational environments to automate structured processes involving multiple participants and decision points. These systems enable documents and information to move between actors in a controlled and traceable manner, improving organizational transparency and coordination [2]. Business process management research further emphasizes the importance of process orchestration in handling complex workflows involving multiple actors and services [3].

Another important paradigm in modern information systems is event-driven architecture (EDA), where system components communicate through asynchronous events rather than tightly coupled service calls. This approach enables flexible and scalable system integration [4].

Service-oriented architecture (SOA) provides a foundation for integrating distributed services through standardized interfaces and messaging mechanisms, enabling interoperability among heterogeneous systems [5]. In addition, automation technologies have been widely adopted to improve service efficiency by reducing manual workload and enabling automated execution of service processes [6].

Despite these technological advancements, existing implementations in higher education institutions often treat conversational systems, workflow management systems, and knowledge sources as separate components. Many chatbot implementations function primarily as information retrieval tools without integration into structured administrative workflows. As a result, academic service requests such as document submissions and approval processes are still handled manually and lack transparency.

To address these challenges, this study proposes an event-driven conversational workflow architecture for academic service automation. The architecture integrates WhatsApp as a conversational interface, a workflow orchestration engine implemented using n8n, and Supabase PostgreSQL as a data management layer. This study advances academic service systems by proposing a unified architecture that integrates conversational interfaces, workflow orchestration, and knowledge management. Specifically, it employs Retrieval-Augmented Generation (RAG) for context-aware responses [12], [13], leverages a low-code, event-driven approach for scalable multi-level approvals [4], [14], and introduces a hybrid interaction model—combining a student chatbot with an administrative dashboard—to balance accessibility and operational control.

The proposed system is validated through a prototype implementation, demonstrating the feasibility of integrating conversational AI, knowledge management, and workflow automation within a unified academic service platform. A primary advantage of the proposed architecture, compared to previous academic chatbots or traditional workflow systems, is its end-to-end integration capability. While conventional chatbots function primarily as static information retrieval tools without initiating backend processes, and traditional workflow engines often lack intuitive user interfaces, this study unifies a RAG-powered conversational interface (WhatsApp) with an event-driven orchestration engine (n8n). This tightly coupled integration ensures that a user's natural language request not only retrieves context-aware answers but also seamlessly triggers, tracks, and completes multi-level administrative workflows without fragmented manual interventions.

II. Related Work

The study of conversational agents has been developed in recent years, especially in the area of digital services and human-computer interaction. Chatbots and other conversational systems allow users to communicate with information systems in natural language, enhancing accessibility and usability [1]. Chatbot-based systems have been used in educational settings to facilitate services to students, such as academic information search and administrative support.

The workflow management systems have been widely researched in the framework of automation of business processes. Through these systems, organizations are able to specify the structure of processes, the roles of participants and decision logic to facilitate complex operational workflows [2]. The research in business process management also emphasizes the significance of process orchestration to coordinate the activities of various actors and services [3]. Flexibility and scalability in process execution is also a priority of modern workflow systems [7].

The study of process modeling has come up with different methods of validating and enhancing workflow design. The workflow models in organizational systems are ensured to be correct, efficient and reliable through metrics and verification methods [8]. The most recent advancements in distributed systems have brought microservices and workflow orchestration models that allow scalable and flexible integration of system components. Microservice-based systems facilitate modular system design, with services loosely coupled and deployable independently [9]. This paradigm is consistent with event-driven architecture, whereby the components of a system interact asynchronously with each other via events [4].

The use of service-oriented architecture (SOA) has been popular in facilitating integration of heterogeneous systems. SOA allows services to interact with each other using standard interfaces, which offers a platform on which workflow can be coordinated and systems can be interoperable [5]. The evolution of intelligent service systems has also been enhanced by the advancement in artificial intelligence. The service systems based on AI incorporate machine learning and data-driven solutions to improve the service delivery process and the decision-making process [10]. Moreover, explainable artificial intelligence (XAI) has been suggested to enhance transparency and trust in AI-based systems [11].

Retrieval-Augmented Generation (RAG) has become an efficient method of enhancing the accuracy of responses in the context of conversational systems through the incorporation of external knowledge sources into the generation process. RAG-based systems are systems that integrate information

retrieval and generative models to generate context-sensitive responses [12], [13]. In addition, low-code development platforms have attracted interest because of their ability to develop systems quickly and integrate several services. These systems enable developers to create workflows in a graphical manner and unify heterogeneous systems with little code development [14].

Despite significant advancements in isolated technologies, very little research has explored how conversational interfaces, knowledge base generation, and workflow orchestration can be seamlessly integrated into a single academic service system. To fill this gap, this paper proposes an event-driven conversational workflow architecture that unifies these elements into a cohesive platform.

III. Research Methodology

The present research follows the Design Science Research (DSR) approach to designing and testing an artifact, which is an academic service automation system. Design Science Research is based on the development and testing of innovative artifacts to address the recognized organizational issues in information systems settings [15].

The study is conducted according to the DSR model, which is divided into a number of steps such as problem identification, objective definition, artifact design, demonstration, and evaluation [16]. The problem identification phase in this study entailed the analysis of the current academic service processes, especially the processes associated with information services and document request processes.

This analysis has led to the following objective of this research, which is to establish a system architecture that can combine conversational interface, knowledge base management, and workflow orchestration to facilitate the automation of academic services. The artifact developed in this study will comprise of three primary workflows, namely, knowledge base generation workflow, information service workflow, and document request workflow. The architecture is event-driven to allow flexible and scalable execution of workflows.

A prototype system was created to illustrate how academic service requests can be triggered using conversational interfaces and handled using automated workflows. The system combines WhatsApp messaging services, n8n workflow automation, and Supabase PostgreSQL as a data layer. The assessment methodology is based on the previous research of the design science, in which prototype-based testing is applied to evaluate the functionality and relevance of artifacts [17]. Moreover, system behavior is analyzed in terms of realistic usage conditions with the help of a case study approach [18].

IV. System Architecture

The proposed system is based on the event-driven architecture embracing conversational interfaces, workflow orchestration, and knowledge base management in a single academic service platform. This design is in line with the current principles of distributed system design which stresses on loose coupling and asynchronous communication among system components [4].

The architecture is divided into four primary layers, which include conversational interface layer, integration layer, workflow orchestration layer, and data layer. These layers collaborate to facilitate three key workflows, including knowledge base creation, information service, and document request processing.

The conversational interface layer allows the users to communicate with the system using WhatsApp messaging services. This layer offers a natural language interaction system, which enables the students to avail academic services in a more intuitive manner [1].

The integration layer links external services, including WhatsApp API, Google Drive, Google Forms, Google Sheets and a web-based dashboard to the workflow system. This layer serves as an interface between user interactions and the internal processes of a system and allows a smooth communication between heterogeneous services. Service integration mechanisms are used in accordance with the principles of service-oriented architecture [5].

The workflow orchestration layer is deployed with the n8n automation platform that handles the execution of the workflow as a result of events caused by a user interaction or system update. Workflow orchestration allows the coordination of many services and actors in a structured process, and it facilitates scalability and flexibility in system design [7], [9]. Supabase PostgreSQL is the data layer that provides the storage of structured data such as academic knowledge, service requests, and workflow states. This central data management solution helps to maintain uniform access to data and effective service monitoring.

Besides the conversational interface, the system has a web-based dashboard to the administrative users. The dashboard allows administrators to handle service requests, do approvals, and upload completed documents. This interface is used to supplement the conversational workflow by offering a structured environment to monitor and control workflow execution.

The use of a web dashboard also improves the usability of the system since it can be used to interact conversely and through a graphical user interface (GUI). This two-sided interface strategy is consistent with the current service system design, in which multiple channels of interaction are combined to enhance the efficiency of operations and user experience. The overall system architecture is illustrated in Fig. 1.

A. Knowledge Base Generation Workflow

Knowledge base generation workflow is in charge of converting the academic documents into structured knowledge that can be accessed by the information service workflow. This workflow assists in creation of a dynamic knowledge base that facilitates retrieval of information in a correct and contextual manner. The workflow is activated when a new document is uploaded to Google Drive. The n8n Google Drive trigger is triggered by the uploaded file and the workflow process is

started. The system then derives textual information in the document and processes the information to determine the relevant academic information including regulations, procedures and service guidelines.

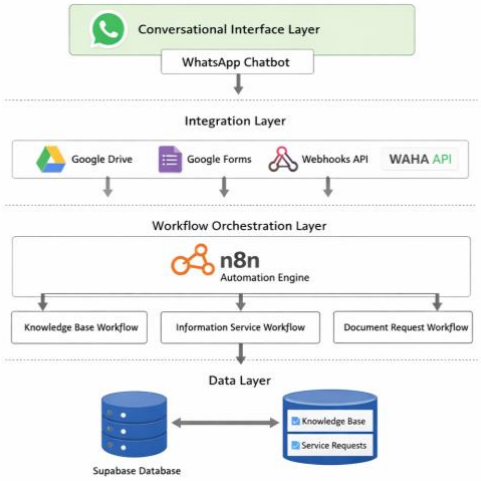

Fig. 1. Unified system architecture of the proposed academic service system

The information that is extracted is saved in the Supabase PostgreSQL database in the form of structured knowledge. This methodology is in line with the recent trends in intelligent service systems, whereby knowledge management based on data is incorporated in service delivery processes.

The knowledge base is constantly updated with the addition of new documents, which guarantees the system with current academic information. The workflow facilitates effective knowledge management and facilitates Retrieval-Augmented Generation (RAG) in the information service workflow. Fig. 2 shows the workflow process.

Fig. 2. Knowledge base generation workflow

B. *Information Service Workflow*

The information service process allows students to access academic data via a chat interface using WhatsApp messaging. This process combines conversational AI and retrieval of knowledge bases to deliver contextual and accurate responses. When a student sends a message via WhatsApp, the message is received by the system via a webhook integration with the WAHA API. An AI agent then interprets the message and conducts intent detection and retrieves information in the knowledge base.

The AI agent is based on a Retrieval-Augmented Generation (RAG) strategy, which is a combination of information retrieval and generative models to generate answers on the basis of external knowledge sources. The method enhances the precision and applicability of the responses as opposed to the conventional chatbot systems.

The information retrieved is utilized to create a response which is relayed back to the student via the WhatsApp messaging interface. This workflow allows the delivery of information in real time and improves the interaction of the users with the academic services.

The combination of conversational interfaces and knowledge-based retrieval systems is a sign of the recent trends in intelligent service systems and AI-based service delivery. The workflow process is illustrated in Fig. 3.

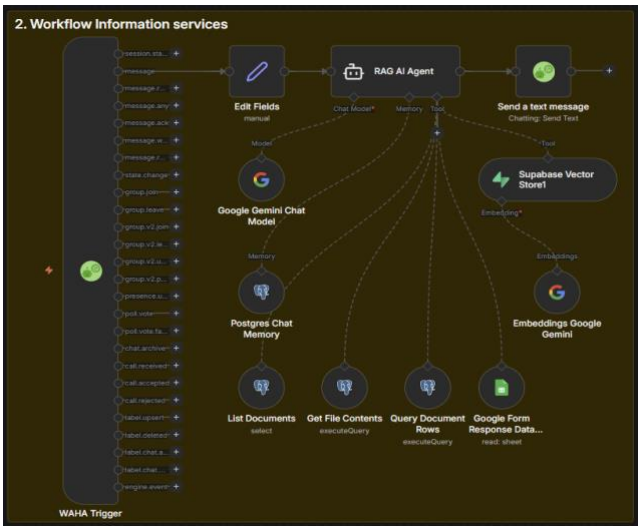

Fig. 3. Information service workflow

C. *Document Request Workflow*

The document request workflow is used to handle academic administrative services that have to be approved in a structured manner. Academic document requests and complaint submissions are some of the services that this workflow supports. Google Forms are used to collect data in a structured form where students post service requests. Every submission will be automatically logged in Google Sheets and initiate an n8n workflow.

The submitted data is stored in the Supabase PostgreSQL database in the workflow to form a record of the service request. The system will then make notifications to the administrative staff and department heads to be approved. The approval process is followed in an event-based manner with each approval action leading to a distinct workflow execution. This will not involve protracted procedures and enhance the efficiency of the system, especially in the case of asynchronous approvals.

The approval process will be divided into two phases: administrative verification and approval of the department head. The orchestration of workflow is used to guarantee that every step is performed in the proper order, which assists in managing the processes in an organized manner. After the request is accepted, the system changes the request status and notifies the student via WhatsApp. This workflow shows how academic administrative workflows can be automated.

Besides WhatsApp-based notifications, the system will also include a web-based dashboard where administrators can review service requests and take approval measures using a graphical interface. The dashboard also has the support of document upload where administrators can upload the finalized document that can be delivered to students using the system. This will enhance the management of workflow and give a better insight into the status of service requests. The interaction with the dashboard and the execution of workflows based on events make the system more flexible and help to conduct the administrative processes more effectively. The workflow process is illustrated in Fig. 4.

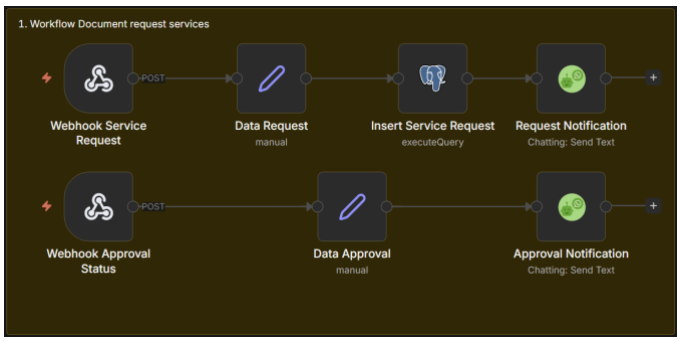

Fig. 4. Document request workflow

The web dashboard interface gives an administrative user a graphical user interface to control academic service workflows. With the help of this dashboard, the administrators will be able to see the incoming service requests, approve them, and attach finalized documents to each request. The Supabase PostgreSQL database is connected to the dashboard and allows updating and retrieving the data on service requests in real time. Every activity carried out in the dashboard like approval or uploading a document initiates respective workflows in the n8n automation platform via API-based communication.

This combination allows the efficient coordination of the dashboard interface and the engine of the workflow orchestration, so that all service processes are implemented in a consistent and efficient way. The dashboard also enhances visibility because the administrators can see the status of every service request such as pending, approved and completed.

The conversational interface is supplemented with the use of a web-based dashboard that offers an organized setting of administrative tasks. Students use the system via WhatsApp, but administrators use the dashboard to better control workflows. This hybrid model of interaction will be more usable and will be in line with current trends in service system design that combine multiple user interfaces.

Moreover, the dashboard facilitates centralized control of the approval process eliminating reliance on message-based interactions and minimizing the chances of approvals being missed. The system balances flexibility and control in the operations of academic services by integrating dashboard-based management and event-driven workflow execution. The dashboard interface, which is a web-based interface, is shown in Fig. 5 and is used to manage the approval processes and document uploads.

V. System Implementation

The n8n workflow automation platform was used as the central orchestration platform to implement the proposed system. Its implementation incorporates several external services, such as WhatsApp messaging via the WAHA API, Google Drive to store documents, Google Forms and Google sheets to submit service requests, and Supabase PostgreSQL as the data storage layer.

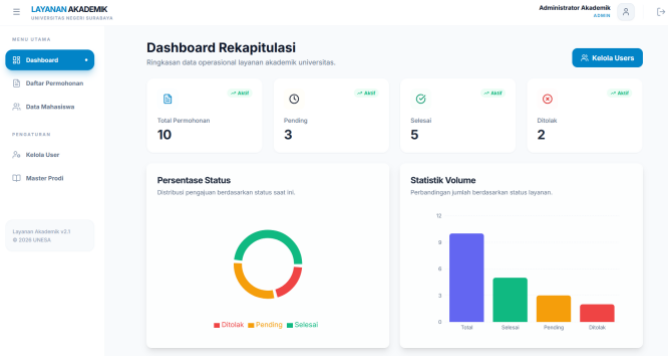

Fig. 5. Web dashboard interface for managing academic service requests, approval processes, and document upload

It is implemented in an event-driven architecture, with every workflow being initiated by certain events (document uploads, incoming messages, or form submissions). This solution enhances the flexibility of the system and allows the service processes to be executed asynchronously. The n8n workflow implementation of the knowledge base generation workflow uses a Google Drive trigger node. A workflow is activated when a new academic document is uploaded, whereby the textual content of the document is extracted and processed to extract relevant information. The extracted data is then saved in the Supabase PostgreSQL database in structured knowledge entries. This process helps in the creation of dynamic knowledge base which may be utilized by the information service workflow.

The information service workflow is deployed with the help of a webhook node where the incoming messages of the WhatsApp API are received. An AI agent processes the messages and detects the intent and retrieves the relevant information in the knowledge base. The system involves a Retrieval-Augmented Generation (RAG) system to produce responses based on retrieved information, enhancing the accuracy of responses and their relevance to the context. The workflow of document request is executed with the help of a trigger node in Google Sheets that identifies new entries in

Google Forms. The workflow puts the data submitted in the database and triggers the approval process. The administration and departmental heads are notified via WhatsApp messaging.

The approval mechanism is applied with the help of triggers based on webhook, where every approval action triggers a different workflow execution. This event-based system does not require long running workflow and is more efficient in the system when dealing with asynchronous approval processes. Moreover, the system follows a modular design of workflow in which every functional element is realized as a workflow. This design enhances maintainability and scalability of the system where individual workflows can be changed without impacting the whole system. This practice aligns with the current workflow orchestration and low-code development paradigms.

A web-based dashboard was also created to assist the administrative functions in the system. The dashboard will have the functions of reviewing service requests, taking approval actions and uploading completed documents. The dashboard communicates with the Supabase database and initiates the running of a workflow by making API calls to the n8n platform. This dashboard is another layer of interaction that is used to supplement the conversational interface. Students communicate with the system via WhatsApp, but the administrators use the dashboard to control workflows more effectively. This hybrid interaction model enhances usability and is consistent with the current system design trends that combine more than one user interface into one system.

System Evaluation

The system evaluation was conducted to assess the functionality and effectiveness of the proposed workflow-based architecture. The evaluation focuses on three main aspects: knowledge base generation, information service interaction, and document request processing. A total of 20 test scenarios were conducted, including 10 information queries, 5 document request submissions, and 5 approval process cases. The results indicate that the system achieved an intent detection accuracy of approximately 90% across the evaluated scenarios. In addition, the average response time for information queries was below 3 seconds (approximately 2.1 seconds), demonstrating the system's capability to support real-time interaction.

For the knowledge base workflow, academic documents were uploaded to Google Drive and processed by the system. The results show that the workflow successfully extracts relevant information and stores it in the database, ensuring that the knowledge base remains accurate and up to date. For the information service workflow, multiple test queries were submitted through the WhatsApp chatbot. The system was able to correctly identify user intent and retrieve relevant information from the knowledge base. The generated responses were consistent with the stored academic data, demonstrating the effectiveness of the Retrieval-Augmented Generation (RAG) approach.

For the document request workflow, test scenarios were conducted by submitting requests through Google Forms. The system successfully processed the requests through multiple stages, including data storage, administrative verification, and department-level approval. The event-driven approval mechanism enabled efficient handling of asynchronous processes.

Overall, the evaluation results indicate that the proposed system effectively integrates conversational interaction, knowledge retrieval, and workflow orchestration within a unified architecture, improving service efficiency and transparency. The assessment outcomes of the proposed system are provided in TABLE 1, where the performance of every workflow component is summarized according to the pre-determined evaluation criteria.

TABLE 1. Evaluation Results of the Proposed System

Scenario	Workflow Component	Evaluation Criteria	Result	Description
Document Upload	Knowledge Base Workflow	Data extraction accuracy	Successful	Documents processed and relevant academic information stored in database
Knowledge Update	Knowledge Base Workflow	Data consistency	Successful	New data added without affecting existing knowledge entries
Information Query	Information Service Workflow	Response relevance	High	System returned accurate and context-aware responses
Intent Detection	Information Service Workflow	Intent classification accuracy	High (90% accuracy)	User intents correctly identified in most test scenarios
Response Time	Information Service Workflow	Response latency	< 3 seconds (avg. 2.1s)	System responded within acceptable time
Document Submission	Document Request Workflow	Data recording accuracy	Successful	Form data correctly stored in database
Admin Approval	Document Request Workflow	Workflow execution	Successful	Approval process executed correctly at admin level
Kaprodi Approval	Document Request Workflow	Multi-level approval	Successful	Sequential approval process executed correctly
Notification Delivery	All Workflows	Message delivery reliability	High	Notifications successfully delivered via WhatsApp
Workflow Execution	Overall System	Process completeness	Successful	All workflows executed without failure during testing

The results of the evaluation provided in Table I show that the proposed system is effective in all the workflow elements. The workflow of knowledge base generation indicates that the data mining is reliable and the knowledge is continuously updated, which means that the system will store the correct and up-to-date academic data. This is necessary in order to facilitate knowledge-based services in the system.

The information service workflow presents high performance with regards to the relevance of response and accuracy in detecting intent. The Retrieval-Augmented Generation (RAG) approach is integrated to allow the system to produce context-based responses on the basis of organized

knowledge to enhance the quality of interaction between the users and the system.

The system is able to manage data submission, multi-level approval and delivery of notification in the document request workflow. The approval mechanism is event-driven, so there is no need to have long-running processes to execute the workflows efficiently; this increases the responsiveness and scalability of the system.

On the whole, the findings indicate that conversational interface, knowledge base management, and workflow orchestration can be used to automate academic services successfully. Not only does the system increase the efficiency of operations, but it also increases the transparency of the services and the experience of the users within the academic setting.

A. Pre- and Post-Automation Workflow Comparison

To quantitatively evaluate the impact of the proposed system, a comparative analysis was conducted between the traditional manual academic service process and the newly implemented event-driven architecture. The evaluation measured key operational metrics, significantly demonstrating the advantages of the automated workflow.

As shown in TABLE 2, the implementation of the event-driven conversational architecture drastically reduces process latency and eliminates manual coordination bottlenecks. Furthermore, user satisfaction is inherently addressed by shifting from an opaque manual process to a transparent, real-time tracking system. The automated workflow provides consistent, 24/7 accessibility for information queries, which was previously constrained by the working hours of administrative staff.

TABLE 2. Pre- and Post-Automation Workflow Comparison

Evaluation Metric	Before Automation (Traditional System)	After Automation (Proposed System)	Improvement / Impact
Information Retrieval Time	Minutes to hours (depending on staff availability)	< 3 seconds (avg. 2.1s)	Real-time response enabled by RAG and AI intent detection.
Service Request Submission	Manual submission via disparate channels (WhatsApp to admin, paper forms)	Centralized structural data collection via Google Forms and WAHA API	Eliminated fragmented requests; unified data logging in Supabase PostgreSQL.
Approval Coordination	Manual follow-ups; high risk of missed messages and process bottlenecks	Automated multi-level event-driven triggers via n8n	Asynchronous, traceable approval execution without manual follow-up.
Process Transparency	Opaque (students cannot track request status)	High visibility (Automated status notifications & centralized Web Dashboard)	Enhanced user experience and administrative tracking.

B. Procedural Comparison Analysis

To evaluate the operational differences brought by the proposed architecture, a comparative analysis was performed against the existing manual academic service framework. This evaluation examines the structural workflow transformation without requiring extensive field integration. Based on this structural comparison, the prototype architecture simplifies the steps required to process academic requests. By replacing manual handovers with automated asynchronous events, the system removes standard procedural bottlenecks inherent in the traditional system.

VI. DISCUSSION

As shown in TABLE 3, the adoption of the proposed system can be viewed as an indication of the possibility of introducing conversational interfaces, workflow orchestration, and knowledge management to academic service settings. The system offers a single platform which facilitates automated information services as well as structured administrative processes.

TABLE 3. Pre- and Post-Automation Workflow Comparison

Process Dimension	Manual Academic Service	Proposed Event-Driven Architecture	Operational Impact
Information Inquiry	Staff-dependent, limited to office hours, manual search.	RAG-powered automated chatbot, available 24/7.	Removes dependency on staff availability for routine information.
Approval Routing	Fragmented via personal chats or physical paper collection.	Automated multi-level triggers via n8n orchestration.	Standardizes the approval path and prevents missed requests.
Data Synchronization	Manual recording across separate spreadsheets or books.	Instant logging into central Supabase PostgreSQL database.	Ensures data consistency and real-time tracking capabilities.

Event-driven architecture enhances the flexibility and scalability of the system because it allows workflows to be executed asynchronously. This method works especially well with approval processes with many actors and time lags. The combination of AI-based conversational systems with the retrieval of the knowledge base increases the interaction of the user and the precision of information delivery. Moreover, low-code platforms like n8n can be used to simplify the development of systems and allow the quick integration of heterogeneous services.

The current implementation has specific limitations, as the system relies on external APIs which may introduce external dependencies. Additionally, the evaluation is based on controlled prototype simulations rather than an institutional-wide production deployment. Consequently, while the initial usability metrics are favorable, actual system performance under heavy user loads remains untested. Future work will focus on integrating this platform with the university's core

academic information system and conducting longitudinal user satisfaction studies in a live campus environment.

A web-based dashboard also adds to the system and gives the administrators more control over the execution of the workflow. In contrast to purely conversational systems, the dashboard allows us to interact in a structured way, which enhances usability and allows more complex administrative functions, including document management and approval tracking.

The future work can involve the incorporation of institutional academic information systems, the development of AI, and user-based testing to determine usability and system performance in the real world. The proposed system also reduces process latency and has a better traceability due to automated workflow coordination compared to the traditional academic service systems that are based on manual coordination.

Compared to conventional academic service systems that rely on manual coordination, the proposed system reduces process latency and improves traceability through automated workflow orchestration. This demonstrates the potential of integrating event-driven architecture with conversational interfaces in academic service environments.

VII. CONCLUSION

This paper presented a conversational workflow architecture based on events to automate academic services. The architecture combines generation of knowledge bases, conversational information services and document request workflow into a single system. The deployment based on the n8n automation platform proves that academic service processes can be automated with the help of event-driven workflows that allow managing the services efficiently and interacting with the users better.

The results of the evaluation show that the system can deliver the correct academic data and manage the administrative services requests with organized approval procedures. The architecture proposed provides a scalable and adaptable model of digital transformation in institutions of higher learning.

VIII. REFERENCES

[1] A. Følstad and P. B. Brandtzæg, “Chatbots and the new world of HCI,” Interactions, vol. 24, no. 4, pp. 38–42, 2017, doi: 10.1145/3085558.

[2] W. M. P. van der Aalst, Workflow Management: Models, Methods, and Systems. Cambridge, MA, USA: MIT Press, 2004.

[3] W. M. P. van der Aalst, A. H. M. ter Hofstede, and M. Weske, “Business process management: A survey,” in Business Process Management, Berlin, Germany: Springer, 2003, pp. 1–12, doi: 10.1007/3-540-44895-0_1.

[4] G. Hohpe and B. Woolf, Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions. Boston, MA, USA: Addison-Wesley, 2003.

[5] T. Erl, Service-Oriented Architecture: Concepts, Technology, and Design. Upper Saddle River, NJ, USA: Prentice Hall, 2005.

[6] M. Lacity and L. Willcocks, “A new approach to automating services,” MIT Sloan Management Review, vol. 58, no. 1, pp. 40–49, 2016.

[7] M. Weske, Business Process Management: Concepts, Languages, Architectures, 2nd ed. Berlin, Germany: Springer, 2019, doi: 10.1007/978-3-662-59432-2.

[8] J. Mendling, Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness. Berlin, Germany: Springer, 2008, doi: 10.1007/978-3-540-89224-6.

[9] O. Zimmermann, N. Stocker, D. Lübke, and U. Zdun, “Microservices tenets and patterns,” IEEE Software, vol. 36, no. 3, pp. 104–107, 2019, doi: 10.1109/MS.2019.2910028.

[10] H. Yang, Z. Chen, and L. Liu, “AI-enabled service systems: A review and future research directions,” Information Systems Frontiers, vol. 23, pp. 1–15, 2021, doi: 10.1007/s10796-020-10045-0.

[11] D. Gunning et al., "XAI: Explainable artificial intelligence," *Science Robotics*, vol. 4, no. 37, 2019, doi: 10.1126/scirobotics.aay7120.

[12] P. Lewis et al., “Retrieval-augmented generation for knowledge-intensive NLP tasks,” in Advances in Neural Information Processing Systems (NeurIPS), 2020.

[13] S. Izacard and E. Grave, “Leveraging passage retrieval with generative models for open-domain question answering,” in Proceedings of the 38th International Conference on Machine Learning (ICML), 2021.

[14] S. Sahay, A. Indamutsa, D. Di Ruscio, and I. Panach, "Supporting the understanding and comparison of low-code development platforms," in *2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2020, pp. 171–178, doi: 10.1109/SEAA51224.2020.00036.

[15] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design science in information systems research,” MIS Quarterly, vol. 28, no. 1, pp. 75–105, 2004, doi: 10.2307/25148625.

[16] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, “A design science research methodology for information systems research,” Journal of Management Information Systems, vol. 24, no. 3, pp. 45–77, 2007, doi: 10.2753/MIS0742-1222240302.

[17] J. Venable, J. Pries-Heje, and R. Baskerville, “FEDS: A framework for evaluation in design science research,” in Proceedings of the 11th International Conference on Design Science Research in Information Systems and Technology (DESRIST), 2016, doi: 10.1007/978-3-319-39237-6_16.

[18] R. K. Yin, Case Study Research and Applications: Design and Methods, 6th ed. Thousand Oaks, CA, USA: Sage Publications, 2018.