Pragmadic: An AI-Driven Platform for Enhancing Digital Nomad Onboarding and Local Integration in Malaysia

Loh Yet Marn, Matthew 1, a),Jayahkudy, Usha 1, b) and S Prabha Kumaresan 2, 3, c)

1School of Computing, INTI International College Penang, 1-Z, Lebuh Bukit Jambul, Bukit Jambul, 11900, Bayan Lepas, Pulau Pinang, Malaysia

2Faculty of Computing and Informatics, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Malaysia.

3Center of Natural Language Processing (NLP), CoE for Artificial Intelligence, Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Malaysia.

*c) Corresponding author: prabha.kumaresan@mmu.edu.my*

*a)matthewloh256@gmail.com  
b)usha.jayahkudy@newinti.edu.my*

**Abstract.** Pragmadic is a web-based platform designed to enhance the onboarding experience of digital nomads (DNs) in Malaysia, particularly applicants of the DE Rantau Nomad Pass visa program. It addresses the challenge of fragmented information and the lack of a centralized resource hub, which hinders seamless integration and access to local opportunities. Pragmadic leverages Retrieval-Augmented Generation (RAG) and large language models (LLMs) to aggregate and deliver real-time, personalized information, streamlining the onboarding process for DNs and supporting coworking space hub owners. Initially implemented in Penang, the platform follows a prototyping methodology for iterative development. The technology stack includes Next.js for the frontend, Supabase for the backend, and AI models from OpenAI, Gemini, and Anthropic. Rigorous functional and non-functional testing, including black-box, integration, and security testing, ensures robust performance and compatibility. The Prototyping model was chosen for its adaptability to evolving stakeholder needs and the dynamic LLM environment. Expected outcomes include reducing information barriers for DNs, fostering local ecosystem connections, and enhancing business visibility within the DE Rantau Hub Partner program through LLM-driven analytics. Pragmadic aims to provide a centralized, intelligent support system that will significantly improve the digital nomad experience in Malaysia.

# INTRODUCTION

The rise of digital nomadism represents a fundamental transformation in work culture and the social contract, enabling professionals to operate independent of location. This trend presents significant economic opportunities for host countries, attracting highly skilled and valued talent while fostering digital economic innovation. However, it also challenges sustainable development across broader social and governance contexts. Malaysia, proactively recognizing this potential, launched the DE Rantau program to establish itself as the preferred digital nomad (DN) hub within ASEAN, providing more permissive visa policies than global competitors [1]. Early statistics from 2024 indicate growing interest, with diverse participating nationalities contributing significantly to the local economy [2].

However, despite increasing concerted investments in the initiative, prospective and current DE Rantau DNs face considerable hurdles. Key challenges include navigating decentralized, fragmented information landscapes when researching visa requirements, accommodation, local regulations (especially taxation), and cultural nuances. Existing platforms, or a lack thereof, lack centralized, personalized, and updated resources, hindering seamless integration and preventing DNs from fully capitalizing on local opportunities, particularly outside major hubs like Kuala Lumpur [3].

To address these gaps, particularly within the context of Penang, an emerging DN destination in Malaysia, this paper introduces the Pragmadic platform, leveraging emerging full-stack frameworks and AI technologies. Pragmadic utilizes Retrieval-Augmented Generation (RAG) for large language models (LLMs) to create a more effective answer engine. The implementation provides DNs with tailored, contextually relevant information from curated DE Rantau program documents. Beyond the RAG core, the platform facilitates onboarding and integration through interactive mapping, event discovery, community networking features, and AI-driven analytics tools for local DE Rantau Hub partners to guide decisions. The overall goal is to present Pragmadic’s position from the standpoint of sustainable development and intelligent computing, employing emergent web and AI technologies to build a more efficient, supportive ecosystem for digital nomads.

# BACKGROUND AND RELATED WORK

While various platforms cater to DNs globally, specific solutions are emerging tailored to national programs like Malaysia's DE Rantau but often exhibit institutional inertia. For instance, the official DE Rantau mobile application (currently only accessible on the Apple App Store), developed in partnership with HostAStay, primarily focuses on booking certified accommodations and lacks comprehensive onboarding support or advanced information retrieval capabilities. Platforms like Citizen Remote offer broader tax, legal, and visa guidance, along with community features across multiple countries. However, they may lack the deep integration with specific local program details needed by DE Rantau participants. Similarly, there exist informational chatbots developed for specific communities, such as university campuses, that rely on earlier NLP techniques or neural networks trained on limited datasets, having potential for more advanced, context-aware systems like RAG for complex domains [4]. The analysis on the existing work of chatbots and DE Rantau related platforms identified a gap for a platform integrating authoritative program information with intelligent, conversational access and localized community/onboarding features.

Addressing the challenge of information retrieval from extensive documentation necessitates advancements relying on the emergent natural language strengths of LLMs. RAG has emerged as a powerful technique to cost-effectively enhance the accuracy and relevance of LLMs [5**Error! Reference source not found.**]. RAG architectures ground LLM responses by first retrieving relevant passages from a specified knowledge corpus before generating an answer. This approach is particularly suited for domains like nomad visa programs, where accuracy and grounding in official sources are paramount. Recent advancements have explored adaptive and self-correcting RAG methods to improve retrieval quality and response generation further [6-7], exploring the potential for more sophisticated conversational agents. A proposed theoretical process flow is depicted in Figure 1(a), which acts as the primary reference of this work.

The development of Pragmadic leverages these concepts synergistically with modern full-stack web architecture. The application of Generative AI and LLMs is expanding into various specialized domains, including personalized healthcare where systems aim to provide tailored advice based on user data [8]. Furthermore, AI techniques, particularly computer vision models like YOLOv5 coupled with tracking algorithms, are increasingly applied to analyze complex real-world data streams, such as monitoring traffic patterns for impact assessment [9]. From a similar standpoint of emergent technology exploration, we utilize LLMs, like OpenAI's API [10], for their state-of-the-art natural language understanding, generation, and embedding capabilities crucial for RAG pipelines and AI analytics. The frontend uses Next.js for robust server-side rendering and cutting-edge React features. For the backend, Supabase provides a full-fledged Backend-as-a-Service (BaaS) solution built on PostgreSQL [11]. Supabase is free and open-source software unifying services for auth, storage, real-time, and crucially, its database supports vector embedding via pgvector. This technical decision streamlines the development of RAG-ready knowledge bases.

|  |  |
| --- | --- |
|  |  |
| (a) | (b) |

**FIGURE 1.** RAG methodology process flow based on adaptive & corrective rag and simplified architectural diagram of pragmadic application layers

# METHODOLOGY AND SYSTEM ARCHITECTURE

The development of the Pragmadic platform followed a Prototyping model consisting of the phases of Communication / Requirements Gathering, Planning, Modelling, Construction, Testing, Deployment and Maintenance, and Feedback and Iteration. The Prototyping software development lifecycle (SDLC) methodology was selected for its inherent flexibility, which is crucial for projects involving rapidly evolving technologies like LLMs and for accommodating emergent user requirements identified through continuous stakeholder feedback. This iterative approach allows for rapid adjustments to both AI features and user interface based on early testing and evolving insights from the dynamic LLM landscape. The methodology is further accelerated with rapid development cycles enabled via utilizing a Supabase-managed backend instance and Vercel's build and deployment platform [12], which provides streamlined CI/CD capabilities, automatic GitHub integration, and enterprise infrastructure features. Adopting Supabase and Vercel for deployment is essential for efficiently iterating and updating AI features.

## Overall System Architecture

Pragmadic has modern, maintainable and scalable full-stack architecture. The key layers are depicted in Figure 1(b):

* Frontend: Developed using Next.js 14 and the newly introduced React 19 Server Components for server-side rendering. The user interface utilizes Shadcn UI (Radix) for accessible and composable components, Framer Motion for animations, and MapboxGL for interactive mapping features. Client-side state management and data fetching are handled using TanStack Query (React Query) for caching and synchronization.
* Backend: Powered by Supabase, providing a PostgreSQL database, user authentication (handling OAuth, magic links, email/password, role-based access control), object storage for documents and user uploads, real-time subscriptions, and serverless Edge Functions. Pgvector is utilized for enabling efficient similarity searches on text embeddings. Drizzle ORM streamlines type-safe database operations and migrations.
* AI Layer: Integrates LLMs primarily from the leading providers (OpenAI, Anthropic, Gemini, Ollama) like the OpenAI APIs (e.g., GPT-4o, text-embedding-3-small). The Vercel AI SDK serves as a crucial abstraction layer, simplifying the implementation of chat management, conversational interfaces, RAG pipelines, and tool-calling functionalities. This SDK manages streaming responses, client-server state synchronization for AI interactions, and provides UI hooks (useChat, useCompletion).
* Deployment: The application is deployed on Vercel to utilize its seamless integration with Next.js, global edge network, serverless functions for API routes, automatic CI/CD pipelines linked to GitHub, and managed infrastructure features. Supabase’s managed instance is also utilized in the interests of resource constraints.

## RAG Answer Engine Implementation

The core of Pragmadic's information retrieval capability lies its RAG implementation, designed to provide accurate answers grounded in official DE Rantau program information. Figure 2(a) shows an answer engine response based on selected documents across various categories. RAG overall is a cost-effective alternative to the expensive fine-tuning process of state-of-the-art foundational models. Additionally, via Ollama or Groq Cloud, open-source LLMs with their tradeoffs and strengths can easily be grounded in context as models continue to be more accessible and competent. The process involves several key stages that are based on the application of the theoretical research background:

* Knowledge Base Creation: Authorized administrators (through RBAC) utilize a dedicated interface within Pragmadic to upload relevant documents (PDF) pertinent to the DE Rantau program. This mirrors approaches in other domain-specific chatbots where maintaining an up-to-date knowledge base via an administrative backend is crucial [5]. These documents are stored securely in Supabase Storage buckets.
* Document Processing Pipeline: A server action triggers an asynchronous processing pipeline upon upload. Documents are parsed (using unpdf), and the extracted text is segmented into smaller, semantically meaningful chunks. Each chunk is then converted into a vector embedding using OpenAI's text-embedding-3-small model. These embeddings and the corresponding text chunks and metadata (source document ID, page number) are stored in a dedicated document\_chunks table, indexed using pgvector for efficient querying.
* Retrieval: When a user submits a query via the chat interface, the query text is embedded using the exact dimensions. A similarity search (specifically, cosine similarity) is performed against the document\_chunks vector store to retrieve the top-K most relevant chunks based on semantic closeness to the user's query. This retrieval process is often enhanced by middleware (ragMiddleware), which may perform steps like query rewriting or hypothetical answer generation to improve retrieval relevance as grounded in theoretical research.
* Generation: The retrieved text chunks are formatted and appended as context to the original user query. This augmented prompt is then sent to an LLM (e.g., GPT-4o) via the Vercel AI SDK's API Next.js endpoint (/api/chat). The SDK handles the interaction with the LLM API, streams the contextually enriched generated response back to the user interface, and manages the conversational state using hooks like useChat.

## AI-Driven Analytics Implementation

|  |  |
| --- | --- |
|  |  |
| (a) | (b) |

**FIGURE 2.** The implementation of RAG Answer Engine / Chatbot Page and AI-Driven Analytics for DE Rantau Hub Owners

For DE Rantau Hub partners, Pragmadic offers an analytics dashboard enhanced with AI capabilities to provide deeper insights into hub activity and member engagement.

* Data Aggregation: The system collects data from user interactions within hubs, including event participation rates and general review/activity metrics. This data is stored in Supabase.
* LLM Tool Calling: The analytics chat interface allows hub owners to pose natural language queries about their hub's performance (e.g., "Show me member demographics," "Analyze recent review sentiment"). These queries are processed by an LLM configured with specific "tools" using the AI SDK's tool-calling features.
* Tool Definition and Execution: Predefined tools (e.g., memberStatsTool, eventStatsTool, reviewAnalysisTool) are implemented as server-side functions. Each tool has a defined schema (using Zod) specifying its expected input parameters. When the LLM determines a tool should be called based on the user's query, the AI SDK facilitates invoking the corresponding backend function with the extracted parameters. These functions execute complex Drizzle ORM queries against the Supabase database to fetch, aggregate, and process the required analytics data.
* Insight Generation and Visualization: The executed tool(s) results are returned to LLM. The LLM then synthesizes this structured data into a natural language summary or directly provides the formatted data needed for visualization. The AI SDK streams this final response back to the analytics chat interface, where insights are displayed textually, and structured data is used to render charts (using Recharts via Shadcn UI) directly within the chat or on the main dashboard. This allows hub owners to interactively explore or get suggestions with their hub data through conversation.
* This methodology combines rapid prototyping with a robust, layered architecture, leveraging modern BaaS and AI SDKs to implement complex features like RAG and LLM agent-driven analytics efficiently.

# IMPLEMENTATION AND RESULTS

This section details the implemented RAG Answer Engine and AI-Driven Analytics features via a prototype deployed on Vercel with a remote Supabase project instance, with pilot quantitative and qualitative evaluations.

## RAG Answer Engine Functionality and Evaluation

The RAG Answer Engine provides users with a conversational interface to query information about the DE Rantau program. Users can select specific documents from the knowledge base, managed by administrators via the file management interface, to scope their queries by categories and/or ask general questions. The system supports multi-turn conversations, maintains context, and allows users to upload images for additional context if needed. Responses are streamed to the UI, providing real-time feedback and an enhanced user interface and user experience.

Black box testing confirmed the core functionality of the RAG module. The achievements include:

* Successful Retrieval: The system correctly identified and retrieved relevant text chunks from the Supabase vector store based on user query embeddings.
* Contextual Generation: LLM responses were successfully grounded in the retrieved context, providing answers relevant to DE Rantau specifics contained within the knowledge base documents.
* Interface Functionality: Document selection, model switching (between different LLM providers like OpenAI), chat history management, error handling, and multi-modal input uploads operated as expected.

Quantitative evaluation was conducted using a comprehensive pilot test suite of 15 questions across four categories: high-priority DE Rantau queries (n=6), medium-priority questions (n=4), edge cases (n=3), and negative tests (n=2). The evaluation methodology employed ground truth validation with verified chunk IDs from the production database, ensuring scientific rigor in the assessment. All tests were conducted using Google's Gemini 2.5 Flash Preview 04-17 model for text generation and OpenAI’s text-embedding-3-small for embedding tasks.

Based on Table 1, the RAG system demonstrated operational reliability with 100% system availability (15/15 tests completed successfully). Retrieval accuracy metrics shows sufficient performance across multiple precision levels: Precision@3 achieved 0.356, Precision@5 reached 0.227, and Precision@10 was 0.180, indicating room for improvement for ranking of relevant chunks due to lack of embedding optimizations. Overall Recall was 0.840, comprehensive coverage of relevant information. The system achieved a Hit Rate of 0.867, retrieving at least one relevant document chunk for 13 out of 15 test queries. Mean Reciprocal Rank (MRR) was 0.656, indicating that relevant chunks typically appeared in the top 2 positions of retrieved results.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TABLE 1.** RAG System Performance Quantitative Evaluation Results | | | | |
| **Category** | **Metric** | **Value** | **Notes** |
| Retrieval Performance | Precision@3  Precision@5  Precision@10  Recall  F1-Score  Hit Rate  MRR  Avg Similarity | 0.356  0.227  0.180  0.840  0.357  0.867  0.656  0.608 | Test case data by categories, benchmarking architecture and testing methodology principles can be accessed from the [repository](https://github.com/matthewloh/pragmadic-v1/blob/update-deps/src/features/benchmark/data/testCases.ts). |
| Generation Quality | Faithfulness  Relevance  Completeness | 4.1 / 5.0  4.1 / 5.0  4.1 / 5.0 | LLM-as-a-Judge-ranked used (Groq Llama 3.3 70B Versatile) |
| System Performance  Reliability | Mean Response Time  Success Rate | 12.1s  100% | ± 3.2s  15/15 completed |

LLM-as-a-Judge evaluation conducted with the cost-effective frontier open-source model Llama 3.3 70B hosted on Groq Cloud yielded high-quality scores across nuanced situations. Faithfulness scored 4.1/5.0, Relevance achieved 4.1/5.0, and Completeness reached 4.1/5.0 (95% confidence intervals). These scores indicate that the system produced accurate, relevant, and comprehensive responses grounded in the retrieved context. F1-Score (harmonic mean of precision and recall) was 0.357, resembling balanced retrieval performance. The average semantic similarity score between queries and retrieved chunks was 0.608, showing meaningful content matching.

## AI-Driven Analytics Functionality and Evaluation

The AI-Driven Analytics module provides hub owners with a standard dashboard and a conversational chat interface for exploring hub data. The dashboard visualizes key metrics such as member role distribution and growth trends using interactive charts (e.g., pie charts, line graphs) rendered with Recharts. The analytics chat interface allows owners to query their data using natural language. The LLM, facilitated by the Vercel AI SDK's tool-calling APIs, interprets these queries and invokes appropriate backend tools. These tools query the database, process the data, and return structured results and/or summaries to the LLM, presenting insights or generating visualizations.

Integration testing focused on the analytics data pipeline and tool-calling functionality. Key findings include:

* Data Flow Integrity: User actions within hubs (e.g., new member joining, event participation) correctly triggered updates that are reflected in the analytics database and subsequently visualized on the dashboard.
* Chart Rendering: Visualization components accurately rendered the processed data fetched from Supabase.
* Tool Calling Efficacy: The LLM successfully identified user intent from natural language queries, invoked the correct backend analytics tools with appropriate parameters, and received structured data back. The tools correctly executed database queries and performed necessary aggregations.

## Security Considerations

Preliminary security testing focused on foundational web security principles and LLM-specific risks as recommended by OWASP [13]. Basic rate limiting was implemented on key API endpoints, particularly the chat API, using Upstash to mitigate simple DoS or abuse patterns. The RAG system's inherent safety and security benefits stem from grounding responses in a controlled knowledge base, reducing hallucination risks in addition to the intrinsic model safety and security properties of commercial LLM APIs. Techniques like multi-step adaptive and corrective RAG processing were implemented from theoretical concepts to enhance accuracy and act as guardrails. RBAC using Supabase Auth and custom JWT claims, restricts access to sensitive functions like knowledge base management and analytics tools. Comprehensive LLM security testing (e.g., advanced prompt injection, model DoS) was beyond the scope of this initial implementation but these foundational measures provide a baseline level of security.

# DISCUSSION

The development of Pragmadic represents a minimum-viable-product of advanced AI techniques like RAG and LLM-driven analytics applied to a platform supporting digital nomads. The prototype successfully addresses the core problem of information fragmentation within the DE Rantau program by providing a conversational, context-aware answer engine grounded in knowledge bases. AI-driven tools also offer a novel approach to extracting valuable, accessible insights for local hub partners, expanding beyond traditional dashboards.

However, several limitations warrant discussion. The platform's broader applicability would require scalability and expansion of the knowledge base and potential localization. The reliance on commercial LLM APIs (e.g., OpenAI) introduces operational costs and dependencies, but architecture allows potential future integration of open-source models. The effectiveness of the RAG relies heavily on the quality and completeness of the ingested documents. Through testing, consistent first party verified accuracy and handling of edge cases was a gap. Larger-scale user studies are required to fully assess usability, user satisfaction, and real-world impact to address potential user gaps, like challenges faced in deploying specialized AI systems in fields like healthcare [14].

Despite limitations, similar to how IoT systems can enhance personal safety and well-being [15], by facilitating skilled worker integration and supporting local partners, Pragmadic serves as a proof-of-concept for leveraging emerging technologies for sustainable development in the context of digital nomadism. The platform facilitates smoother integration of skilled professionals with tools for local ecosystem partners. This can contribute to knowledge transfer, local economic stimulation, and the overall sustainable growth of regional digital economies like Penang's.

# CONCLUSION

This paper introduces Pragmadic, a web platform designed to enhance the digital nomad experience in Penang, Malaysia, by addressing critical information access and integration challenges within the DE Rantau program. The implementation is of a RAG-integrated answer engine and AI-driven analytics, leveraging LLMs and modern web technologies. Initial evaluations confirmed the system’s feasibility and core functionality in providing grounded information retrieval and rich data insights. Pragmadic offers a pathway to support the sustainable development of local digital economies through improved resource accessibility and integration tools for the growing digital nomad community. Future work should expand into external non-technical factors, conducting comprehensive user validation studies, and exploring open-source LLM solutions to enhance decentralization, scalability, and security.

# REFERENCES

1. J. Bednorz, “Working from anywhere? Work from here! Approaches to attract digital nomads,” Annals of Tourism Research **105**, 103715 (2024).
2. W. Tracz, editor , “Domain-specific software architecture (DSSA) frequently asked questions (FAQ),” SIGSOFT Softw. Eng. Notes **19**(2), 52–56 (1994).
3. J. Antony Xavier, and Z.U. Ahmad, “Proposed scholarly research agenda for transforming Malaysia into a model developing nation,” International Journal of Public Sector Management **25**(3), 231–243 (2012).
4. M. Bonet, and D. Fernández-Quijada, “Sounds without borders: Exploring the cross-national expansion of commercial European Radio Groups,” European Journal of Communication **36**(6), 610–625 (2021).
5. T.-J. Goh, L.-Y. Chong, S.-C. Chong, and P.-Y. Goh, “A Campus-based Chatbot System using Natural Language Processing and Neural Network,” JIWE **3**(1), 96–116 (2024).
6. H. Larochelle and Neural Information Processing Systems Foundation, editors , *34th Conference on Neural Information Processing Systems (NeurIPS 2020): Online, 6-12 December 2020* (Curran Associates, Inc, Red Hook, NY, 2021).
7. S. Jeong, J. Baek, S. Cho, S.J. Hwang, and J. Park, “Adaptive-RAG: Learning to Adapt Retrieval-Augmented Large Language Models through Question Complexity,” in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, (Association for Computational Linguistics, Mexico City, Mexico, 2024), pp. 7036–7050.
8. R. Lakatos, E.K. Urbán, Z.J. Szabó, J. Pozsga, E. Csernai, and A. Hajdu, “Designing Prompts and Creating Cleaned Scientific Text for Retrieval Augmented Generation for More Precise Responses from Generative Large Language Models,” in *2024 IEEE 3rd Conference on Information Technology and Data Science (CITDS)*, (IEEE, Debrecen, Hungary, 2024), pp. 1–6.
9. M.T.-T. Yong, S.-B. Ho, and C.-H. Tan, “Migraine Generative Artificial Intelligence based on Mobile Personalized Healthcare,” *Journal of Informatics and Web Engineering* **4**(1), 275–291 (2025).
10. J.J. Ng, K.O.M. Goh, and C. Tee, “Traffic Impact Assessment System using Yolov5 and ByteTrack,” *Journal of Informatics and Web Engineering* **2**(2), 168–188 (2023).
11. British Standards Institution., International Organization for Standardization., and Technical Committee ISO/TC 46, Information and documentation., *Information and Documentation: Work Process Analysis for Records = Information et Documentation : Analyse Du Processus Des Records.*, 1st ed., 2008-06-15. (BSI, [London], 2008).
12. Kyiv National University of Construction and Architecture, 31 Povitroflotskyi Avenue, Kyiv, 03037, Ukraine, O. Kozakova, I. Kravchenko, M. Sulayman, D. Kuśnierz-Krupa, Cracow University of Technology, Faculty of Architecture, 24 Warszawska Street, 31-155, Cracow, Poland, S. Wang, M. Abdulgani Mustafa, M. Lisińska-Kuśnierz, Cracow University of Economics, College of Management and Quality Sciences, 27 Rakowicka Street, 31-510, Cracow, Poland, L. Bednarz, Wroclaw University of Science and Technology, Faculty of Civil Engineering, 27 Wybrzeze Stanislawa Wyspienskiego Street, 50-370, Wroclaw, Poland, M. Budziakowski, and Cracow University of Technology, Faculty of Architecture, 24 Warszawska Street, 31-155, Cracow, Poland, “The Role of Photographic Documentation in the Process of Conservation of Destroyed Architectural Monuments and Centres of Historic Cities,” Int J Conserv Sci **15**(SI), 3–16 (2024).
13. C. Gales and Splunk, Inc., *The Product Is Docs: Writing Technical Documentation in a Product Development Group*, Second edition. (publisher not identified, [Place of Publication Not Identified], 2020).
14. S. Yusif, and A. Hafeez-Baig, “A Conceptual Model for Cybersecurity Governance,” *Journal of Applied Security Research* **16**(4), 490–513 (2021).
15. R. K, and G.T. W, “IoT-Based Nerve Stimulator for Women’s Safety,” JIWE **4**(1), 129–139 (2025).