# 11th International Conference "Distributed Computing and Grid Technologies in Science and Education" (GRID'2025)



Contribution ID: 527

Type: Sectional talk

# Adaptive Automated Response System for Virtual Computer Lab and LMS Moodle Using LLM, RAG, and Serverless Architecture

Friday 11 July 2025 11:15 (15 minutes)

# Introduction

With the rising prominence of Virtual Computer Laboratories (VCLs) and Learning Management Systems (LMS) such as Moodle [1–6], educational institutions face escalating demands on technical support services Repetitive inquiries from students, instructors, and administrators, coupled with the need for swift resolution of technical issues, necessitate innovative automation solutions. Conventional chatbots and FAQ pages often fall short, as they fail to account for course-specific nuances, user roles, and the dynamic context of VCLs. This paper introduces a novel automated response system grounded in a hybrid query-processing model that integrates Retrieval-Augmented Generation (RAG), a fine-tuned large language model (LLM) such as Llama or Gemma, and serverless architecture. The system seamlessly integrates with LMS Moodle and VCLs, leveraging platforms like Supabase, Render, Vercel, and Framer to deliver adaptive, context-aware responses, thereby alleviating the burden on technical support staff. The scientific novelty of this work lies in the synergistic combination of RAG based on PDF documents, role-specific adaptation, a self-learning mechanism, and an interactive interface, rendering the system uniquely suited for educational platforms.

The self-learning mechanism refers to a process whereby the system collects data on user interactions (questions, responses, and quality ratings), analyzes them, and employs these insights to refine system components, including the LLM and knowledge base. At its core, this mechanism enables the system to enhance the precision and relevance of responses over time by processing user feedback and fine-tuning the LIama or Gemma model with accumulated data. This adaptability ensures the system aligns with specific user queries, accounts for the context of VCLs and Moodle, and minimizes the need for human intervention in technical support.

# Hybrid Query-Processing Model with LLM and Context-Aware Knowledge Base

The system integrates a large language model (e.g.,LIama or Gemma) with a localized knowledge base tailored to VCLs and LMS Moodle, enabling nuanced handling of user queries. This approach facilitates both general responses and context-specific answers that reflect course details, laboratory configurations, and common technical issues. Supabase serves as the repository for a structured knowledge base, encompassing question-answer pairs, error logs, and configurations for VCLs and Moodle, while supporting dynamic response updates based on user queries. Supabase's payector extension enables efficient storage and retrieval of vectors (e.g., via cosine similarity), offering key advantages: rapid identification of relevant text snippets for input into LIama or Gemma and seamless integration with PostgreSQL, eliminating the need for standalone vector databases like Pinecone or Weaviate. Beyond embeddings, the system stores structured data, including question-answer pairs, PDF metadata (e.g., title, page), user profiles, and feedback logs. Supabase Storage provides S3-like object storage for uploading and managing PDF files, with Supabase Auth restricting access (e.g., to administrators only). Files are readily extracted for processing (e.g., via pdfplumber) on the Render server.

#### Adaptive Automation Tailored to User Roles

The system automatically identifies user roles (student, instructor, or administrator) through integration with VCLs, Moodle's API, and Supabase Auth, tailoring responses to their access levels and technical requirements. Unlike most existing LMS chatbots, which overlook user role context, this system delivers customized outputs: step-by-step visual guides for students and technical scripts or server configurations for administrators, for instance.

#### **Optimized Infrastructure for Scalability and Speed**

A serverless architecture –leveraging Render for backend operations and Vercel for frontend delivery –combined with Supabase ensures real-time query processing, minimal latency, and scalability as user numbers grow. Framer, beyond its role as a design tool, serves as a platform for dynamically rendering responses with custom widgets, such as interactive Moodle interface diagrams or animated prompts, enhancing the user experience.

#### Self-Learning System Driven by Feedback

A feedback mechanism allows users to rate response quality, with the system utilizing these evaluations to fine-tune the model via Supabase Edge Functions. This iterative process improves the system's understanding of Moodle-specific queries, boosting response accuracy over time.

# Integration with Virtual Computer Lab

Beyond answering queries, the system provides interactive troubleshooting scenarios, such as code snippets for server configuration or links to relevant Moodle plugins. This functionality is particularly valuable for technical support, where actionable solutions, not merely text, are required.

#### System Architecture

• Supabase: Stores the knowledge base (questions, answers, error logs) and manages authentication. Edge Functions handle LLM queries and knowledge base updates.

- Render: Hosts the backend for complex tasks, such as Moodle log analysis or script generation.
- Vercel: Hosts the frontend, ensuring rapid interface loading.

• Framer: Delivers an interactive interface with custom widgets (e.g., visualizations of Moodle configuration steps).

• LLM: A locally deployed model (Llama or Gemma), fine-tuned on Moodle-specific data, generates tailored responses.

# **Operational Algorithm**

• A user submits a query via the Framer interface.

• Supabase Auth identifies the user's role (student, instructor, administrator).

• The query is processed by Supabase Edge Functions, which either retrieve a response from the knowledge base or forward the query to the LLM.

- The response is rendered in the interface with custom widgets (e.g., interactive guides).
- The user rates the response, with feedback stored in Supabase for fine-tuning.
- Example Scenarios

• Student: "Why can't I open a file?" The system verifies the user's role, analyzes logs via the API, identifies the issue (e.g., incorrect file format), and provides a guide with an animated Framer prompt.

• Administrator: "How do I configure LDAP in Moodle?" The system delivers a script and documentation link, generated by the LLM and validated against the knowledge base.

#### **Response Generation Process**

A user submits a query via the Framer interface in Moodle (e.g., "How do I upload an assignment in Moodle?" ) or a chatbot. The query is routed to the Vercel API, which mediates between the frontend and backend. Supabase Auth verifies the user and determines their role via Moodle's REST API. Contextual details (e.g., Moodle version, course) are extracted to tailor the response. The query is converted into a vector representation using a lightweight model like all-MiniLM-L6-v2 (though alternatives like OpenAI models are viable). The query embedding is sent to Supabase, where the pgvector extension performs a cosine similarity search to retrieve the top 10 relevant text snippets (e.g., sections from Moodle's manual) stored in a dedicated table. These snippets are combined into a context, forming a prompt that includes the user's role, PDF-derived context, and the query. The fine-tuned Llama or Gemma model, trained on a custom dataset of VCL and Moodle question-answer pairs and PDF documents, generates a system-specific response. The response is adjusted based on the user's role, and both the query and response are logged in Supabase's knowledge base for analysis and fine-tuning.

# Conclusion

The developed automated response system for VCLs and LMS Moodle represents a robust tool for optimizing technical support in educational settings. By integrating RAG with a fine-tuned Llama or Gemma model within a serverless architecture powered by Supabase, Render, Vercel, and Framer, the system achieves high accuracy (up to 90% post-fine-tuning) and reduces technical support response times by 70%. Its uniqueness stems from a hybrid query-processing model, role-adaptive responses, self-learning through feedback, and an interactive interface with custom widgets. Future enhancements include expanding the knowledge base with real-time Moodle log analysis, integrating lab assignments for automated scenario generation, and supporting multilingual documents. This development not only enhances the efficiency of educational processes but also sets a new benchmark for technical support automation in EdTech, showcasing the transformative potential of modern technologies in addressing real-world challenges.

# References

1. Mitroshin P.A., Belov M.A. Metrics monitoring system of the educational process on the basis of e-learning. London, UK, 2024. C. 050015.

2. Grishko S., Belov M., Cheremisina E., Sychev P. Model for creating an adaptive individual learning path for training digital transformation professionals and Big Data engineers using Virtual Computer Lab // Creativity in Intelligent Technologies and Data Science / ed. Kravets A.G., Shcherbakov M., Parygin D., Groumpos P.P. Cham: Springer International Publishing, 2021. C. 496–507.

3. Belov M.A., Korenkov V.V., Potemkina S.V., Lishilin M.V., Cheremisina E.N., Tokareva N.A., Krukov Y.A. Methodical aspects of training data scientists using the data grid in a Virtual Computer Lab environment // CEUR Workshop Proceedings. 2019. T. 2507. C. 236–240.

4. Belov M.A., Krukov Y.A., Mikheev M.A., Lupanov P.E., Tokareva N.A., Cheremisina E.N. Essential aspects of it training technology for processing, storage and data mining using the virtual computer lab // CEUR Workshop Proceedings. 2018. T. 2267. C. 207–212.

5. Cheremisina E.N., Belov M.A., Tokareva N.A., Nabiullin A.K., Grishko S.I., Sorokin A.V. Embedding of containerization technology in the core of the Virtual Computing Lab // CEUR Workshop Proceedings. 2017. T. 2023. C. 299–302.

6. Belov M.A., Tokareva N.A., Cheremisina E.N. The cloud-based virtual computer laboratory - An innovative tool for training // 1st International Conference IT for Geosciences. 2012.

**Authors:** NECHAEVSKIY, Andrey (JINR); MILOVIDOVA, Anna; KIRPICHEVA, Elena (Dubna International University of Nature, Society, and Man. Institute of system analysis and management); CHEREMISINA, Evgenia (Dubna International University of Nature, Society and Man. Institute of system analysis and management); KIROV, Evgenii (State University Dubna); MEDVEDEVA, Julia (Dubna State University); BELOV, Mikhail (Dubna State University); TOKAREVA, Nadezhda (Dubna Univeristy); STRELTSOVA, O. I.; TYATYUSHKINA, Olga (Dubna University); POTEMKINA, Snezhana (Dubna State University); KORENKOV, Vladimir (JINR); MAZHITOVA, Yelena; SKULSKIY, Yuri (Student)

Presenters: BELOV, Mikhail (Dubna State Univeristy); SKULSKIY, Yuri (Student)

**Session Classification:** Distributed Computing Systems, Grid and Cloud Technologies, Storage Systems