Design and Implementation of an Intelligent SQL Learning Platform for Computer Engineering Students

Jorge Luis Cruz Agosto
Master in Computer Engineering
Advisor: Alfredo Cruz, Ph.D.
Polytechnic University of Puerto Rico
Graduate Project EXPO, February 2026

Abstract *— Database skills are a core competency for computer engineering students that they will need in their careers after graduation. In this era of Big Data, those with strong database skills will have an advantage. However, databases are a broad topic, so they cannot be covered in a single class. This may leave students with only basic knowledge and lingering doubts, especially if the instructor lacks the right tools. To enhance the students' learning experience, this paper proposed a learning website that would help students better understand databases and feature an artificial intelligence assistant that would act as a virtual tutor. The virtual tutor would support students by answering questions, clarifying doubts, and providing tailored explanations. It is important to note that the project did not aim to replace teachers. This combination of personalized content, practice exercises, and adaptive tutoring created a customized learning environment that enhanced engagement and knowledge retention.*

Keywords *— Artificial Intelligence, Database Education, eLearning, Interactive Learning.*

INTRODUCTION

In our digital world today, data plays a crucial role in almost every industry. This makes it essential for computer engineers to be skilled in designing, managing, and optimizing databases. Mastery of databases allows professionals to structure and query data efficiently and make informed decisions that influence system performance, reliability, and cost savings. For instance, well-optimized queries and carefully designed schemas can greatly reduce computational load and storage needs, leading to significant resource savings for companies. Additionally, in the era of Big Data, where organizations deal with vast amounts of structured, unstructured, and semi-structured data that continues to grow exponentially over time [1], those with strong database skills gain a competitive edge.

E-learning refers to the way educational content is delivered through digital platforms, giving learners the flexibility to access materials whenever and wherever they choose, whether asynchronously or in real-time, on various devices. Its adoption has grown by nearly 900% since the early 2000s, reflecting a global shift toward mobile learning, a trend that has been further accelerated by the COVID-19 pandemic [2]. Mobile learning also has pedagogical benefits such as personalization, scalability, and ongoing feedback. A systematic review of the literature on mobile learning conducted between 2014 and 2023 identified four key benefits: increased flexibility, enhanced learning effectiveness, greater intrinsic motivation, and higher student engagement. Additionally, the review also highlighted that mobile learning encourages active participation and fosters autonomy, supporting student-centered pedagogies [3]. A study developed a web-based e-learning system for database courses that integrates SQL lessons, quizzes, and an ERD simulator, and found strong acceptance and received a good score when assessed using ISO/IEC 25010 software quality evaluation criteria [4].

LITERATURE REVIEW

As previously mentioned, mobile learning adoption has been increasing recently. Mobile devices like phones and tablets are now more common and the go-to for many young people, including students. In addition to improving access, evidence shows that e-learning systems lead to significant gains in student motivation and, as a result, better academic performance. The authors

also highlight the importance of interactive features, suggesting that the effectiveness of e-learning environments is closely tied to their ability to sustain learner interest [5]. A systematic review of mobile learning research confirmed that mobile-based approaches have a notably positive effect on academic achievement compared to traditional methods. Beyond performance, the study highlighted how mobile learning fosters greater interaction, collaboration, and critical thinking skills among students. The portability and accessibility of mobile devices also enable more personalized learning experiences, supporting student engagement outside the classroom [6]. Another research did a systematic review of 50 empirical studies [7]. The findings revealed that the adoption of mobile learning had a positive impact on various areas, including Computer Science [7]. However, they also noted that having the technology is not enough. Educators must also become familiar with technology-enhanced learning environments early in their training and foster an environment that favors pedagogical innovation, adapting to the specific needs of students and the educational context [7]. This was also demonstrated in another study, which showed that the adoption of interactive technologies in online university courses correlates with significant improvements in student motivation and knowledge retention, while also highlighting key implementation barriers, such as insufficient instructor training [8].

To maximize these benefits of mobile learning, adaptivity is crucial, and artificial intelligence can improve the adaptability of educational tools by addressing learners' needs, leveraging their strengths, and guiding them toward mastery. Research also highlighted that AI has become a crucial part of the modern academic landscape. While it is true that there are some risks, such as the accuracy of information, there are good practices to mitigate those [9]. Additionally, a study carried out with 40 students found positive correlations between the use of generative AI tools and the success of solving database administration problems during a practical exam [10]. Artificial Intelligence has the potential to enhance a technology's capacity to accommodate students' needs, leverage their strengths, and foster their knowledge and skills [11]. A systematic review found that interventions using ChatGPT significantly improved university-level students' academic performance, affective-motivational states, and higher-order thinking skills, while also reducing mental effort [12]. It was also noted in that review that AI models, such as ChatGPT, are effective in language education because they can articulate ideas clearly [12]. Lastly, another systematic literature review of 63 studies on the integration of AI/ML-driven adaptive learning in e-learning environments found that personalized content delivery, real-time feedback, and dynamic learning paths significantly increase student engagement and academic performance [13].

METHODOLOGY

The development of the DataMentor platform followed a design-based research approach aimed at addressing the gap between theoretical database instruction and hands-on practical learning. Rather than an empirical study, this project focuses on the design and implementation of an educational system informed by pedagogical literature and grounded in modern web technologies. The methodology integrates principles from e-learning, mobile learning, and adaptive instruction to create a platform capable of providing students with a self-paced, interactive, and personalized educational experience.

Currently, numerous educational websites address database topics. However, not all are tailored for users with minimal to no experience in the field, which is a common case for students. At this early stage, it is crucial that students also comprehend the purpose of each concept and their applicable scenarios. Currently, many students often complete these courses with residual doubts and without the required experience to apply database skills effectively in the actual industry [14]. Secondly, not all websites adequately test the student's knowledge. A good approach to this is by following Bloom's

Taxonomy, as it fosters students' engagement in the classroom and stimulates critical thinking ability and problem-solving skills in learners [15].

The design of the website is key to this effort and should consider elements like color and font choices. The website features a clean and modern design intended to capture the interest of young computer engineering students while making navigation easy. The student can choose between light or dark theme, where both will feature a technology-oriented aesthetic suitable for the platform's target audience. The interface prioritizes simplicity by having all user actions reachable with one click. A systematic review of 25 recent studies was conducted and found that strong usability, intuitive interface design, and a user-centered experience in online learning platforms significantly affect students' satisfaction, engagement, and ultimately learning outcomes, thus highlighting that poor UX/UI is a barrier to effective e-learning [16].

Ultimately, it is universally acknowledged that each student exhibits unique strengths and weaknesses in their learning process. While educators are responsible for facilitating understanding across diverse topics for each student, they may be constrained by limited time or resources, which prevents them from creating compelling learning experiences. Now, in the era of artificial intelligence, we can give educational websites the capability to provide this personalized experience to the students. Although there are some challenges, a study found that AI-based intelligent tutoring systems (ITS) can deliver personalized, adaptive instruction and produce learning gains comparable to human tutoring in structured domains [17]. This was also confirmed by a systematic review, which found that AI-based systems in higher education significantly enhance personalized learning, student engagement, and administrative efficiency, despite the challenges [18].

The objective of this project was to develop an educational platform designed to help computer engineering students in their database courses by providing an interactive and adaptive learning experience. The platform was designed to offer learning modules that promote self-paced learning. The students begin with theoretical lessons, then reinforce their understanding through practice exercises, and finally apply those skills in a real SQL sandbox environment. Additionally, an AI-powered virtual tutor is available to provide guidance and clarification when needed, helping students learn about databases on the platform. This cyclical approach of learn, practice, and apply mirrors effective instructional design principles and helps students build confidence as they progress through the modules. While the platform was designed for computer engineering students, it can be beneficial for anyone seeking to enhance or refresh their database skills. A visual representation of this learning flow is provided in Figure 1, illustrating how students engage with the platform's features in a structured and meaningful manner.

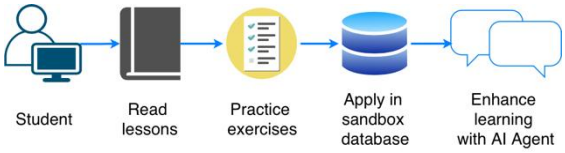

Figure 1
Learning Low Diagram

IMPLEMENTATION

This section describes the technical implementation of the DataMentor platform, focusing on the architectural decisions, system components, and technologies used to support its functionality. The goal of the implementation was to translate the pedagogical design described in the previous section into a scalable, secure, and responsive web-based learning environment.

System Architecture

The DataMentor platform was designed using a modular architecture that integrates modern web technologies, cloud services, and artificial intelligence to support an interactive learning experience. Figure 2 provides an overview of the system architecture and the communication between its main components. The DataMentor platform was created to provide a reliable, easy-to-use educational

experience for computer engineering students. It combines modern web tools, cloud services, and artificial intelligence to offer a complete learning experience that includes theoretical content, practice exercises, and personalized tutoring.

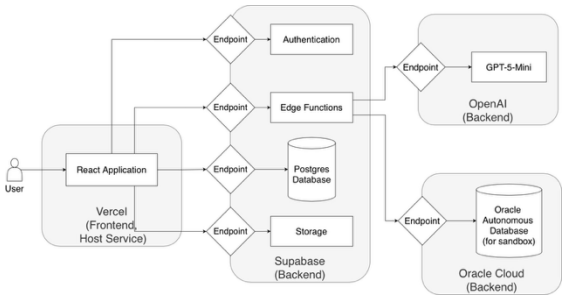

Figure 2
Project Architecture

The client-facing part of the system is a React-based frontend, chosen for its modular structure, responsiveness, and strong community support. This interface is styled using Tailwind CSS and hosted on Vercel, which is a platform that offers automatic deployments, global content delivery, and serverless scalability. These features ensure that students can access the platform seamlessly across different devices and operating systems, whether they are using a desktop computer or a mobile phone.

The backend of the application is powered by Supabase, an open-source Backend-as-a-Service solution that provides multiple services such as authentication, database management, and file storage. Supabase Auth handles user registration, login, and password recovery, while its PostgreSQL database stores metadata related to modules, lessons, exercises, and student progress. The platform also leverages Supabase edge functions to securely communicate with external services, including the Oracle Autonomous Database and OpenAI's GPT-5 Mini API, keeping secret values on the server side. Lesson content is stored in Supabase buckets in Markdown format, with references maintained in the database to indicate where to fetch it from the front end.

To provide hands-on learning, the platform includes a SQL sandbox environment hosted on Oracle Cloud. This sandbox allows students to practice SQL queries in a safe and isolated space. When a student initiates a sandbox session, a personal schema is created under their user ID, ensuring that any changes made do not affect other users. This feature is particularly valuable for reinforcing concepts taught in the lessons and for providing real-world experience with Oracle SQL.

Artificial intelligence was included to enhance the learning experience of the website. The platform integrates OpenAI's GPT-5 Mini model to serve as a virtual tutor that can answer questions, clarify doubts, and offer guidance. The AI agent is located in both the lesson view and the sidebar, and it is designed to respond in the student's preferred language. It defaults to Oracle SQL when the dialect is unclear and provides concise explanations followed by relevant code examples.

Security was also taken into consideration during the system's design. Supabase implements row-level security policies to ensure that users can only access their own progress data. Authentication tokens are required for database access, and permissions were assigned as read-only for most tables. The AI agent is also configured with safety protocols to avoid misinformation and to communicate uncertainty when necessary.

The architecture supports horizontal scalability and high performance. Vercel's serverless infrastructure automatically adjusts to varying traffic loads, while Supabase's real-time database capabilities allow for responsive interactions. The use of edge functions reduces latency by executing logic close to the user's location. The Oracle Autonomous database supporting the sandbox can also scale to meet demand.

Mobile accessibility is a key consideration in the platform's design. The responsive layout ensures that students can engage with lessons, exercises, and the AI tutor from any device, from smartphones to laptops. This flexibility supports self-paced learning, making the platform more inclusive and adaptable to various learning environments.

User Interface and Features

Figure 3 shows the Learning Content home screen, which contains all the available lessons. A

course directory is located on the left, displaying lessons organized by module. For any completed lesson, there will be a green check mark after the name. On the right side of the course directory, all the lessons will be shown in card format. Each card contains the lesson's name, the difficulty, the description, and to which module it belongs. The website includes six modules in total, each containing lessons related to the topic. The initial five modules focus on relational databases using Oracle, while the concluding module will provide a brief overview of non-relational databases. This last module provides the student with a suggestion of what to learn next.

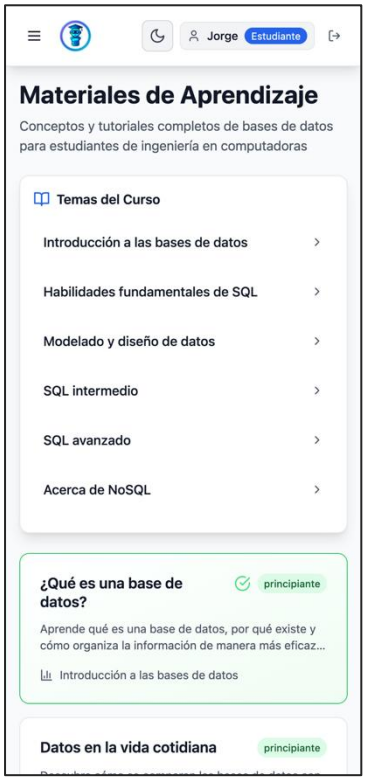

Figure 3
Learning Content Screen

Each lesson follows a structured flow designed to make each topic easier to understand. The format begins by presenting the problem, which answers the question of why the topic matters and sets the context for learning. This is followed by the theory section, where the concept is explained in detail, addressing what the topic is about. Next, students may encounter, if applicable to the lesson, some SQL code examples that demonstrate the practical application of the theory. These examples are written in Oracle SQL, aligning with the platform's focus and ensuring consistency with the sandbox environment. To further enhance the learning experience, each lesson includes a dedicated section titled "Tips from the Experts", which offers professional insights into best practices or common pitfalls to avoid. Finally, every lesson concludes with a summary recap, which reinforces the key takeaways of the lesson and helps students consolidate their understanding before moving on to practice exercises or the next lesson. Figure 4 illustrates how the lesson content is presented across desktop and mobile views.

Figure 4
Lesson Theory Screen

Each lesson within the platform includes a set of multiple-choice practice exercises designed to reinforce the concepts covered in the theory section, as seen in Figure 5. These exercises were made following Bloom's Taxonomy, beginning with basic recall questions and progressing toward more complex tasks that require analysis and application. This pedagogical approach ensures that students not only memorize SQL syntax but also develop the ability to apply it in realistic scenarios. Additionally, the website can generate additional practice questions based on the current lesson through the use of AI. Generating questions uses tokens, and the feature will be unavailable when the user runs out of tokens until the next cycle. All questions will indicate whether the student's choice was correct and explain the correct answer.

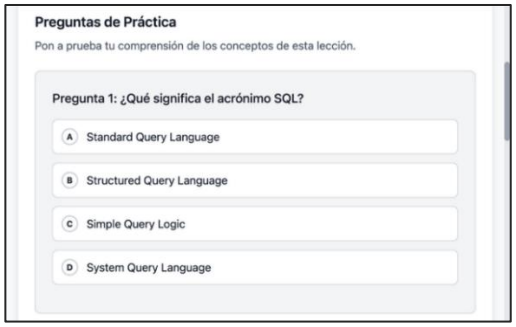

Figure 5
Lesson Practice Exercise Screen

If the students have any doubts while reading a lesson, they can click on the AI button to open the chat box, as seen in Figure 6, and ask the AI tutor their question. The AI agent already has a preset context that tells it will act as a database tutor, answering Oracle SQL questions and using the same language as the student. In addition, a hint is also passed to the AI agent indicating which lesson the student is reading. This additional context enhances the accuracy of the response. Any message received from the agent is rendered in Markdown format, improving the readability of generated code examples. Each prompt consumes tokens, and once the student has used all of them, the system will not allow any more prompts. Chat history only lasts during that session and is not preserved in the database. Similarly, information from the student is not shared with the AI agent. Only the prompt is sent, and the database stores only the number of tokens used.

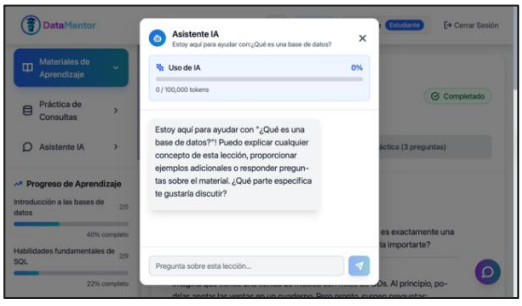

Figure 6
AI Agent Pop-up Screen

The SQL sandbox is the second feature of the sidebar that enables students to apply what they've learned in a safe, hands-on environment. It is powered by an Oracle Autonomous Database hosted on Oracle Cloud and designed to simulate real-world database interactions without the risk of affecting shared data. This sandbox helps students avoid setting up a local Oracle database on their computer for practice and provides an option for those whose study device is a phone or tablet. When a student initiates a sandbox session, the system creates a new temporary user in the database and a new personal schema with their user ID. The temporary user created does not have access to any other table, it only has access to the ones under the user's schema. This approach allows students to experiment freely, make mistakes, and learn through trial and error, which is essential for mastering database concepts.

The sandbox interface is intuitive and straightforward. It includes a schema panel displaying available tables and columns in the sandbox database, a query editor where students can write and execute SQL statements, and a results area that shows query outputs in real time. Additionally, the page provides ready-to-use SQL examples to help students get started and understand common query patterns. The Figure 7 illustrate the layout and functionality of the sandbox across desktop and mobile views, including examples of successful queries and error handling.

Figure 7
Sandbox Database Screen

DISCUSSION

The development of this project resulted in a database learning platform that combines theoretical content, practical exercises, and artificial intelligence to support computer engineering students in their database courses. This website will

allow users to learn or refresh their knowledge of database concepts for Oracle SQL. Some advantages of implementing the platform with React include excellent performance, responsiveness, and compatibility across different devices, which are essential when students are the target audience. Additionally, using Supabase as the backend provided seamless integration of authentication, database management, and progress tracking, enhancing the reliability of the system. Lastly, using Oracle Autonomous Database allows for supporting a varying number of simultaneous users over time in a cost-effective manner, as the database instance will scale up and down to match demand. This automatic scaling feature was needed as the sandbox is one of the main features of the platform.

The website was designed to provide a complete learning experience for the students using the platform. Users can first read the learning modules to learn the database concepts. Then, they can test their knowledge with a few practice exercises. Lastly, they can try a hands-on experience by using the sandbox, powered by Oracle Autonomous Database. Even when the sandbox has some limitations in what the students can do, it still provides real experience with an Oracle database and avoids them having to set up a local instance. If the student has any doubts or needs further clarification during this process, they can use the AI agent, which serves as a virtual tutor within the platform. The interface design was made with visual elements such as progress indicators, colors that reduce eye fatigue, and easy navigation to keep the student engaged.

CONCLUSION

The goal of this project was to develop a tool that would help computer engineering students learn more easily the database concepts. The outcome of this tool was an educational website that featured a responsive design so that students could use it regardless of the device they were using. The platform combines theoretical modules, practical exercises, and a sandbox environment for hands-on practice. Additionally, an AI assistant that serves as a virtual tutor was added to clarify students' doubts or provide more information about the topic. By using React and Supabase, the platform offers good performance, scalability, and resiliency, thus increasing the website's availability.

Material for the lessons was taken from online sources and processed through the GPT-5 model, developed by OpenAI, to use concise and action-oriented language that promotes student engagement. Lessons were written in Markdown format to leverage the code formatting when presenting SQL syntax and queries, as it automatically highlights the keywords of the language. The Markdown format also supports the inclusion of tables, which were utilized in several lesson examples to present the output of SQL queries clearly. Currently, all lessons on relational databases focus on Oracle SQL, while the last module discusses NoSQL in general.

Nowadays, data is generated constantly from our devices, cloud-based applications, and IoT devices. This continuous stream of information makes it essential for computer engineering students to learn how to manage, structure, and analyze data effectively through database systems. Understanding databases prepares students with the tools necessary to store, retrieve, and optimize data. As we move deeper into the era of Big Data, where organizations handle massive volumes of structured and unstructured information, strong database skills become critical. However, it is important to note that students should not learn databases only if they are pursuing a career in Big Data or as a database administrator. Even if they are interested in a career as a software developer, knowing those skills will give them a significant advantage.

As it is, the educational platform DataMentor represents a useful resource for anyone interested in learning about databases. With further development, such as implementing the suggested changes in the previous section, universities could adopt DataMentor as a learning resource for their students to complement the existing database courses, especially since most design decisions were made with cost in mind. Ultimately, this project represents

a meaningful step toward bridging the gap between academic theory and practical database skills.

FUTURE WORK

The website has met the initial requirements set during the scope of this project. However, its capabilities can be further enhanced with the following suggested improvements. Expanding the content library with additional modules could make the curriculum more comprehensive by covering more specific database topics or exploring other SQL-like databases, such as MySQL, SQL Server, or PostgreSQL. Another area that can be improved is how modules are stored. Currently, module files are manually stored in a Supabase bucket, which requires developer intervention to add or update content. Adding a dynamic content management system would make the platform more flexible and easier to maintain. Another area to improve the website is language support, as the interface and all lessons are currently available only in Spanish. Supporting different languages would help make the platform accessible to a broader range of students across various regions.

Additionally, the AI assistant could also be improved. One way to achieve this is by leveraging Supabase's support for retrieval-augmented generation (RAG), which could enable the system to store and retrieve smaller chunks of module content, resulting in more contextually accurate responses. Another approach is to support more than one model and allow switching between them. Currently, the assistant runs on GPT-5 Mini, from OpenAI. However, implementing a mechanism to switch dynamically to the more powerful GPT-5 model for complex queries would improve the quality of the answers. Alternatively, the switch can also be to models from other providers, such as Gemini made by Google.

REFERENCES

[1] Google Cloud. (n. d.). *What is big data?* [Online]. Available: https://cloud.google.com/learn/what-is-big-data. [Accessed: Aug. 19, 2025].

[2] Coursera Staff. (2024, May 9). *What is e-learning?* [Online]. Available: https://www.coursera.org/gb/articles/e-learning. [Accessed: Aug. 18, 2025].

[3] J. Kang, "Benefits and challenges of mobile-learning brought to student learning outcomes in higher education: A systematic review from 2014–2023," in *International Journal of Academic Research in Progressive Education and Development*, vol. 13, no. 1, pp. 20698, 2024. Available: https://doi.org/10.6007/IJARPED/v13-i1/20698.

[4] A. P. M. Dela Rosa, L. M. M. Villanueva, J. M. R. San Miguel, and J. E. B. Quinto, "Web-based Database Courses e-Learning Application," in *Int. J. Computing Sciences Research*, vol. 7, pp. 1531-1543, Aug. 2022. Available: https://doi.org/10.25147/ijcsr.2017.001.1.115.

[5] F. Yahiaoui et al., "The impact of e-learning systems on motivating students and enhancing their outcomes during COVID-19: A mixed-method approach," Frontiers in Psychology, vol. 13, pp. 874181, 2022. Available: https://doi.org/10.3389/fpsyg.2022.874181.

[6] K. Demir and E. Akpınar, "The effect of mobile learning applications on students' academic achievement and attitudes toward mobile learning," in *Malaysian Online Journal of Educational Technology*, vol. 6, no. 2, pp. 48–59, 2018.

[7] L. Pedraja-Rejas, C. Muñoz-Fritis, E. Rodríguez-Ponce, and D. Laroze, "Mobile learning and its effect on learning outcomes and critical thinking: A Systematic Review," in *Applied Sciences*, vol. 14, no. 19, pp. 9105, 2024. Available: https://doi.org/10.3390/app14199105.

[8] J. Guaña-Moya, Y. Arteaga-Alcívar, S. Criollo-C, and D. Cajamarca-Carrazco, "Use of Interactive Technologies to Increase Motivation in University Online Courses," in *Educ. Sci.*, vol. 14, no. 12, Art. 1406, 2024. Available: https://doi.org/10.3390/educsci14121406.

[9] A. M. Vieriu and G. Petrea, "The impact of artificial intelligence (AI) on students' academic development," in *Education Sciences*, vol. 15, no. 3, pp. 343, 2025. Available: https://doi.org/10.3390/educsci15030343.

[10] D. López-Fernández and R. Vergaz, "ChatGPT in Computer Science Education: A Case Study on a Database Administration Course," in *Applied Sciences*, vol. 15, no. 2, pp. 985, 2025. Available: https://doi.org/10.3390/app15020985.

[11] U.S. Department of Education, Office of Educational Technology, Artificial intelligence and the future of teaching and learning: Insights and recommendations, U.S. Department of Education, 2023. [Online]. Available: https://www.ed.gov/sites/ed/files/documents/ai-report/ai-report.pdf.

[12] R. Deng, M. Jiang, X. Yu, Y. Lu, and S. Liu, "Does ChatGPT Enhance Student Learning? A Systematic Review

and Meta-Analysis of Experimental Studies," in *Computers & Education*, vol. 227, art. 105224, 2025. Available: https://doi.org/10.1016/j.compedu.2024.105224.

[13] I. Gligorea, M. Cioca, R. Oancea, A.-T. Gorski, H. Gorski, and P. Tudorache, "Adaptive Learning Using Artificial Intelligence in e-Learning: A Literature Review," in *Educ. Sci.*, vol. 13, no. 12, art. 1216, 2023. Available: https://doi.org/10.3390/educsci13121216.

[14] M. Ishaq, A. Abid, M. S. Farooq, M. F. Manzoor, U. Farooq, K. Abid, and M. A. Helou, "Advances in database systems education: Methods, tools, curricula, and way forward," in *Education and Information Technologies*, vol. 28, no. 3, pp. 2681–2725, 2023. Available: https://doi.org/10.1007/s10639-022-11293-0

[15] S. Nurmatova and M. Altun, "A comprehensive review of Bloom's Taxonomy integration to enhancing novice EFL educators' pedagogical impact," in *Arab World English Journals*, vol. 14, no. 3, Oct. 2023. [Online]. Available: https://ssrn.com/abstract=4593934.

[16] T. K. Miya and I. Govender, "UX/UI design of online learning platforms and their impact on learning: A review," in *Int. J. Research in Business & Social Science*, vol. 11, no. 10, pp. 316–327, Dec. 2022. Available: https://doi.org/10.20525/ijrbs.v11i10.2236.

[17] M. Zerkouk, M. Mihoubi, and B. Chikhaoui, "A Comprehensive Review of AI-based Intelligent Tutoring Systems: Applications and Challenges," arXiv, arXiv:2507.18882 [cs.IR], Jul. 2025. Available: https://doi.org/10.48550/arXiv.2507.18882.

[18] C. Merino-Campos, "The Impact of Artificial Intelligence on Personalized Learning in Higher Education: A Systematic Review," in *Trends in Higher Education*, vol. 4, no. 2, art. 17, 2025. Available: https://doi.org/10.3390/higheredu4020017.