iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 Issue: IV Month of publication: April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79365

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

Smart Vocabulary Builder: An AI-Driven Personalized Learning System Using Spaced Repetition and Generative AI

Dr. C. Udhaya Shankar[1], Elamurugan R[2], Lochana T[3]
[1]Assistant Professor, Department of Artificial Intelligence and Data Science, SNS College of Engineering, Coimbatore, Tamil Nadu, India
[2, 3]Bachelor of Technology, Department of Artificial Intelligence and Data Science, SNS College of Engineering, Coimbatore, Tamil Nadu, India

Abstract: The acquisition and long-term retention of new vocabulary are frequently impeded by traditional, passive learning methods like rote memorization, which fail to address the cognitive principles of memory and contextual understanding. This project, the Smart Vocabulary Builder, presents a full-stack, AI-powered web application designed to overcome these challenges by creating a highly efficient, personalized, and interactive learning environment. The system is architected with a robust Python-based FastAPI backend, a dynamic React.js frontend, and leverages a database managed by SQLAlchemy. At its core, the application implements a Spaced Repetition System (SRS), utilizing a priority queue to intelligently schedule word reviews at optimal intervals, thus combating the natural forgetting curve. Innovation within the project is driven by the deep integration of the Google Gemini API, which transforms the learning process into a conversational experience. When a user adds a new word, the application initiates an "AI Consultation," providing not just a definition, but also a contextual example sentence and a creative mnemonic. This interaction is dynamic, allowing users to regenerate explanations if clarity is not achieved. The platform further personalizes the learning journey by proactively suggesting new, relevant words based on the user's existing vocabulary. A key feature is the integrated, browser-native "Listen & Practice" tool, which combines text-to-speech and speech-to-text functionalities for pronunciation feedback without reliance on costly external services. The result is a feature-complete, intelligent learning platform that successfully merges fundamental computer science algorithms, modern full-stack development practices, and the power of generative AI to offer a demonstrably more effective and engaging alternative to conventional vocabulary-building methods.

Keywords: Artificial Intelligence, Spaced Repetition System, Vocabulary Learning, Generative AI, Personalized Learning, FastAPI, React.js, Natural Language Processing.

I. INTRODUCTION

Vocabulary acquisition is a critical component of language proficiency, directly influencing reading comprehension, writing ability, and effective communication. Despite its importance, traditional vocabulary learning approaches—such as rote memorization and static dictionary usage—remain inefficient, often leading to rapid forgetting and limited contextual understanding. These methods fail to incorporate cognitive principles of memory retention and do not support active recall or personalized learning, resulting in suboptimal outcomes for learners. Recent advancements in intelligent learning systems have introduced adaptive techniques to address these limitations. In particular, Spaced Repetition Systems (SRS) have been widely recognized for their ability to enhance long-term memory retention by scheduling reviews at optimal intervals based on user performance. Concurrently, the emergence of Generative Artificial Intelligence has enabled the dynamic generation of contextual content, including definitions, example sentences, and mnemonic aids, thereby improving comprehension and engagement. However, existing vocabulary learning platforms often treat these technologies in isolation, lacking a unified framework that integrates adaptive algorithms with AI-driven content generation and interactive learning features. To address these challenges, this paper proposes the Smart Vocabulary Builder, an AI-driven, full-stack web application that combines Spaced Repetition techniques with Generative AI to create a personalized and efficient vocabulary learning environment. The system leverages a FastAPI-based backend, a React.js frontend, and a SQL-based database architecture to deliver scalable and responsive performance. It incorporates AI-powered word explanations, personalized vocabulary recommendations, and speech-based pronunciation practice using Text-to-Speech (TTS) and Speech-to-Text (STT) technologies.

II. EXISTING SYSTEM

Current vocabulary learning systems include flashcard-based tools, online dictionaries, and language learning applications. Flashcard platforms such as Anki and Quizlet support memorization and, in some cases, implement Spaced Repetition techniques to improve retention. However, these systems largely depend on static, user-generated content, which may lack consistency and contextual depth. Online dictionaries provide accurate definitions and example usages but function primarily as reference tools without supporting active learning or personalized progress tracking. Language learning applications like Duolingo and Memrise incorporate gamification to enhance user engagement, but they often focus on basic vocabulary and predefined learning paths, limiting flexibility for advanced or personalized learning. Additionally, these platforms provide limited contextual explanations and lack dynamic content generation capabilities. Overall, existing systems do not offer a unified solution that combines adaptive learning, contextual understanding, pronunciation practice, and intelligent content generation. This results in fragmented learning experiences and reduced effectiveness in long-term vocabulary retention.

A. *Key Limitations of Existing E-commerce Systems*

1) Lack of Personalization: Most existing vocabulary learning systems follow a generalized approach and do not adapt to individual learner behavior, performance, or progress. As a result, users receive the same content regardless of their learning pace or proficiency level, leading to inefficient learning outcomes and reduced engagement.
2) Limited Contextual Understanding: Many platforms provide only static definitions and basic example sentences, which are often insufficient for understanding real-world usage. Without dynamic, context-rich explanations, learners struggle to grasp nuances, multiple meanings, and appropriate application of words in different scenarios.
3) Passive Learning Approach: A significant number of systems rely heavily on rote memorization techniques, where learners passively review content without active interaction. This approach does not effectively engage cognitive processes such as recall, reinforcement, and application, resulting in poor long-term retention.
4) Fragmented Learning Experience: Existing tools often separate key learning components such as vocabulary storage, pronunciation practice, contextual learning, and review mechanisms. This lack of integration forces users to rely on multiple platforms, creating a disjointed and less efficient learning experience.

B. *Objective*

The primary objective of this project is to develop an intelligent, AI-driven vocabulary learning system that enhances retention, contextual understanding, and pronunciation through personalized and adaptive learning techniques. The system aims to integrate a Spaced Repetition System (SRS) with Generative AI to provide dynamic explanations, optimized review scheduling, and interactive learning features. Additionally, it seeks to create a unified platform that supports vocabulary management, speech-based practice, and personalized word recommendations, thereby improving overall learning efficiency and user engagement.

1) To enhance vocabulary retention: To design and implement a Spaced Repetition System (SRS) that schedules word reviews at optimal intervals, ensuring improved long-term memory retention and reducing the forgetting rate.
2) To provide contextual understanding of words: To enable learners to understand vocabulary in real-world contexts by generating dynamic definitions, example sentences, and usage scenarios using Generative AI.
3) To develop a personalized learning system: To create an adaptive learning environment that analyzes user performance and behavior to tailor vocabulary suggestions and learning pathways.
4) To improve pronunciation skills: To incorporate speech-based learning features using Text-to-Speech (TTS) and Speech-to-Text (STT) technologies for accurate pronunciation practice and feedback.
5) To provide intelligent vocabulary recommendations: To design a recommendation system that suggests relevant new words based on the user's existing vocabulary and learning progress.

C. *Scope*

1) Development of an Intelligent Vocabulary Learning System: The project focuses on designing a web-based platform that enables users to add, manage, and learn vocabulary efficiently through an adaptive and user-centric approach, improving both retention and usability.
2) Implementation of Spaced Repetition Techniques: The system incorporates a Spaced Repetition System (SRS) to schedule reviews at optimized intervals based on user performance, ensuring effective long-term memory retention and minimizing forgetting.

3) Integration of Generative AI for Contextual Learning: The application leverages Generative AI to provide dynamic definitions, contextual example sentences, and mnemonic aids, enhancing understanding and making learning more interactive and meaningful.
4) Interactive Pronunciation and User Engagement Features: The scope includes speech-based functionalities using Text-to-Speech (TTS) and Speech-to-Text (STT) technologies, allowing users to practice pronunciation and receive real-time feedback within the platform.
5) Scalable Full-Stack Architecture and Future Extensibility: The system is developed using modern technologies such as FastAPI, React.js, and SQL-based databases, ensuring scalability and flexibility to support future enhancements like gamification, analytics, and mobile application integration.

III. SOFTWARE REQUIRMENTS

A. Programming Language

The development of the Smart Vocabulary Builder system involves the use of multiple programming languages to support both backend and frontend functionalities. Python is used as the primary backend programming language due to its simplicity, scalability, and strong support for web frameworks and AI integration. It is utilized in conjunction with FastAPI to build high-performance APIs and handle server-side logic.

For frontend development, JavaScript is employed along with the React.js framework to create a dynamic, responsive, and user-friendly interface. Additionally, HTML and CSS are used for structuring and styling the web application, ensuring compatibility across modern browsers.

B. System Requirements

1) RAM: 4GB or 8GB
2) Windows 10
3) Intel Core i5/i7 processor
4) At least 60 GB of Usable Hard Disk Space

C. Libraries and Frameworks

1) FastAPI: FastAPI is a modern, high-performance Python web framework used for building backend APIs. It supports asynchronous programming, enabling faster request handling and improved scalability. FastAPI provides automatic validation using Pydantic models and generates interactive API documentation. In this project, it is used to manage server-side logic, user authentication, and communication between frontend and backend.
2) SQLAlchemy: SQLAlchemy is a powerful Object Relational Mapper (ORM) used for database management in Python applications. It allows developers to interact with the database using Python objects instead of raw SQL queries. In this system, SQLAlchemy is used to handle data storage, retrieval, and management of user vocabulary, ensuring efficient and secure database operations.
3) React.js: React.js is a popular JavaScript library used for building dynamic and responsive user interfaces. It uses a component-based architecture, allowing reusable UI components and efficient state management. In this project, React.js is used to create an interactive frontend, enabling smooth user experience and real-time updates.
4) Generative AI API: Generative AI APIs are used to provide intelligent, dynamic content such as word definitions, example sentences, and mnemonic aids. These APIs enhance the learning experience by offering context-aware explanations and personalized suggestions. In this system, AI integration plays a key role in improving user engagement and understanding.
5) JWT Authentication (JSON Web Tokens): JWT is used for secure user authentication and session management. It ensures that only authorized users can access the system by generating and validating tokens for each session. This enhances the security and reliability of the application

D. System Integration and Testing

The Smart Vocabulary Builder is designed using a modular and layered full-stack architecture, where multiple components are seamlessly integrated to function as a cohesive system. The frontend, developed using React.js, interacts with the backend through RESTful APIs built using FastAPI, enabling efficient data communication and real-time user interaction. The backend handles core functionalities such as user authentication, vocabulary management, review scheduling, and AI integration.

SQLAlchemy is utilized as the Object Relational Mapper (ORM) to facilitate structured and efficient database operations, ensuring reliable storage and retrieval of user data. The system also integrates Generative AI services to dynamically generate contextual definitions, example sentences, and mnemonic aids, enhancing the overall learning experience. Additionally, JWT-based authentication is implemented to provide secure access control and session management. The integration of Web Speech APIs enables speech-based interaction, allowing users to practice pronunciation using Text-to-Speech and Speech-to-Text functionalities. This tightly coupled integration ensures that all components operate in synchronization, delivering a responsive, scalable, and user-centric application.

Testing of the system is conducted using a comprehensive multi-level strategy to ensure functionality, reliability, and performance. Unit testing is performed on individual modules, including API endpoints, database operations, and authentication mechanisms, to verify correctness at the component level. Integration testing is carried out to ensure proper interaction between the frontend and backend, validating data flow, API responses, and system interoperability. Functional testing evaluates key features such as word addition, AI-generated explanations, spaced repetition scheduling, and pronunciation feedback to ensure they meet specified requirements. System testing is conducted to assess the complete application under real-world scenarios, ensuring stability and performance under varying conditions. Furthermore, User Acceptance Testing (UAT) is performed to gather feedback on usability, interface design, and overall learning effectiveness. Performance testing is also considered to evaluate response time and scalability, especially during concurrent user interactions. The testing results indicate that the system performs reliably, maintains data integrity, and successfully fulfills its objective of providing an intelligent and efficient vocabulary learning platform.

To further ensure robustness and maintainability, the system follows a modular design approach with clearly defined interfaces between components, enabling easier debugging, testing, and future enhancements. Error handling and validation mechanisms are implemented at both frontend and backend levels to prevent invalid inputs and ensure system stability. Continuous integration practices can be adopted to automate testing and deployment processes, improving development efficiency and reliability. Additionally, security testing is considered to identify vulnerabilities in authentication and data handling, ensuring the protection of user information. The system is also designed to support scalability, allowing it to handle increasing user loads and data volume without performance degradation. These practices collectively contribute to building a resilient, secure, and production-ready application.

IV. IDEATE

A. *Proposed System*

1) Requirement Collection and Preprocessing: The requirement collection phase is conducted using a systematic analysis of user needs, domain constraints, and limitations of existing vocabulary learning systems. Requirements are derived through comparative study of current platforms, user behavior analysis, and identification of key challenges such as inefficient retention, lack of contextual learning, and absence of adaptive mechanisms. The system requirements are categorized into functional and non-functional specifications. Functional requirements include modules for user authentication, vocabulary management, AI-driven content generation, spaced repetition scheduling, and speech-based interaction. Non-functional requirements address system scalability, low-latency response, data consistency, usability, and security constraints, ensuring reliable system performance under varying workloads. In the preprocessing phase, the collected requirements are formalized and transformed into structured system specifications. Input data, particularly vocabulary entries, undergo normalization processes such as case standardization, token validation, and duplicate elimination to maintain data integrity. Preprocessing also includes defining data schemas using relational models, mapping entities such as users, words, and review schedules. API contract definitions are established to standardize communication between frontend and backend components. Additionally, input prompts for Generative AI modules are preprocessed to ensure consistent formatting, context relevance, and efficient response generation. This stage also involves workflow modeling, including state transitions for spaced repetition and user interaction flows.
2) AI-Powered Word Explanation: This module utilizes Generative AI to provide dynamic and context-aware explanations for vocabulary words. It generates definitions, example sentences, and mnemonic aids to enhance understanding. The system allows regeneration of explanations, enabling users to explore multiple interpretations until clarity is achieved.
3) Spaced Repetition System (SRS): The system implements an adaptive SRS algorithm to optimize learning intervals based on user performance. Each word is assigned a difficulty level and review schedule, which is dynamically updated depending on correct or incorrect responses. This approach minimizes forgetting and improves long-term retention.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

4) Vocabulary Management Module: This module enables users to add, store, and manage vocabulary efficiently. Each word entry includes attributes such as definition, difficulty level, and next review date. The system ensures structured data handling and easy retrieval for review sessions.
5) AI-Based Word Recommendation: The system provides personalized vocabulary suggestions based on the user's existing knowledge and learning patterns. By analyzing stored vocabulary data, the AI recommends relevant and progressive words, ensuring continuous and guided learning.
6) Cost Estimation and Budget Planning: The cost estimation and budget planning for the Smart Vocabulary Builder focus on minimizing development and operational expenses while ensuring system scalability and performance. The development phase primarily utilizes open-source technologies such as Python, FastAPI, React.js, and SQLAlchemy, which significantly reduce software licensing costs. The major cost components are associated with cloud infrastructure, AI API usage, and deployment services. The integration of Generative AI services (e.g., Gemini API) introduces variable costs based on the number of API requests and usage volume. To optimize these costs, the system can implement efficient request handling techniques such as caching frequently accessed responses and limiting unnecessary API calls. Additionally, lightweight database solutions and scalable cloud hosting platforms (such as AWS, GCP, or Azure) can be used to balance performance and cost.
7) Review and Learning Workflow: The system follows a structured workflow where users add vocabulary, receive AI-generated explanations, and engage in review sessions based on SRS scheduling. Feedback from users continuously updates the learning model, ensuring adaptive and efficient learning.

B. *Advantages*

The proposed Smart Vocabulary Builder offers several advantages over traditional vocabulary learning systems by integrating adaptive learning techniques and modern technologies. It provides a personalized learning experience by analyzing user performance and tailoring vocabulary suggestions and review schedules accordingly. The implementation of the Spaced Repetition System (SRS) enhances long-term retention by optimizing review intervals and reducing forgetting. The integration of Generative AI enables dynamic and context-rich explanations, including definitions, example sentences, and mnemonic aids, which improve comprehension and practical usage of words. Additionally, the system incorporates pronunciation practice using Text-to-Speech and Speech-to-Text technologies, allowing users to develop accurate speaking skills. By combining vocabulary management, contextual learning, pronunciation, and review mechanisms into a single platform, the system ensures a unified and engaging learning experience. Furthermore, the use of scalable technologies such as FastAPI and React.js enables efficient performance and supports future enhancements.

V. RESULT AND SCREENSHOTS

A. *Output*

Fig 1: Output Page 1

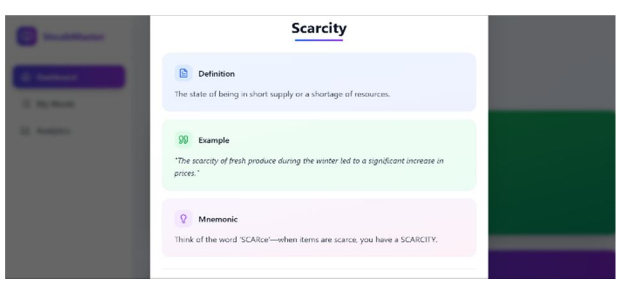

Fig 2: Output Page 2

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

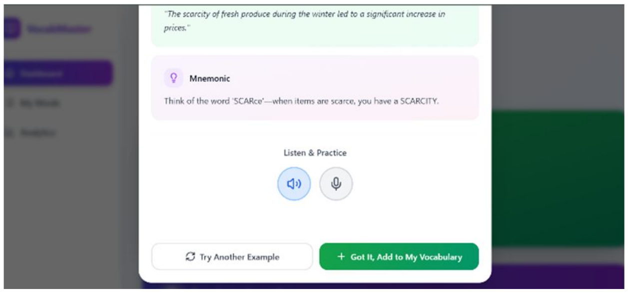

Fig 3: Output Page 3

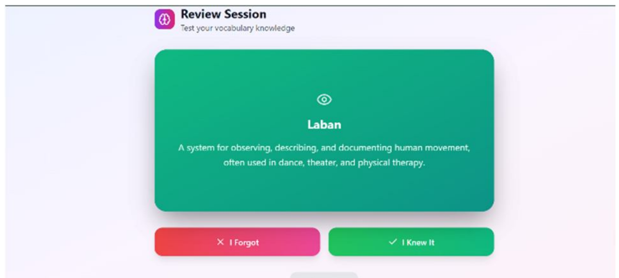

Fig 4: Output Page 4

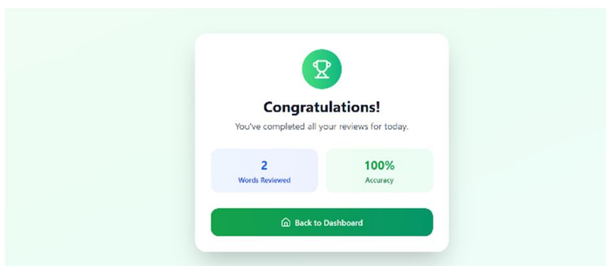

Fig 5: Output Page 5

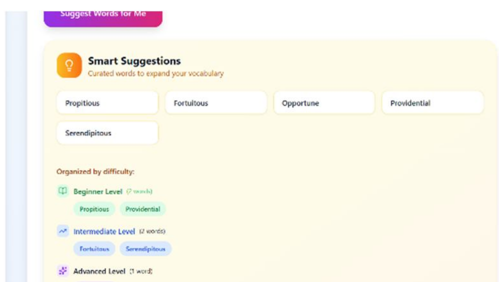

Fig 6: Output Page 6

VI. CONCLUSION

The Smart Vocabulary Builder presents an effective solution to the limitations of traditional vocabulary learning methods by integrating Artificial Intelligence with adaptive learning techniques. The system successfully combines a Spaced Repetition System (SRS) with Generative AI to enhance vocabulary retention, contextual understanding, and pronunciation skills. By providing personalized learning experiences, dynamic content generation, and interactive features, the platform transforms vocabulary acquisition into an engaging and efficient process. The implementation using a full-stack architecture ensures scalability, performance, and seamless user interaction. Overall, the proposed system demonstrates the potential of AI-driven educational tools in improving learning outcomes and offers a strong foundation for future advancements such as gamification, advanced analytics, and mobile application integration.

VII. FUTURE SCOPE

The future scope of the Smart Vocabulary Builder focuses on enhancing user engagement and learning effectiveness through advanced features and technologies. One potential improvement is the integration of gamification elements such as achievement badges, leaderboards, and daily streaks to motivate consistent usage and improve user retention. Additionally, the system can be extended to include AI-generated quizzes, adaptive assessments, and visual learning aids such as image-based mnemonics, catering to different learning styles. Expanding the AI capabilities to provide deeper semantic analysis and contextual variations of words can further enhance comprehension and real-world applicability.

From a technical perspective, the system can be scaled by developing dedicated mobile applications for Android and iOS platforms, enabling users to access the platform anytime and anywhere. The integration of advanced analytics can provide insights into user performance, learning patterns, and progress tracking, allowing for more precise personalization. Furthermore, migrating to distributed cloud infrastructure and incorporating offline capabilities can improve system reliability and accessibility. These enhancements will transform the platform into a comprehensive, scalable, and intelligent learning ecosystem.

REFERENCES

[1] FastAPI Documentation, "FastAPI Framework for Building APIs," Available: HTTPS://FASTAPI.TIANGOLO.COM/
[2] React Documentation, "React: A JavaScript Library for Building User Interfaces," Available: https://react.dev/
[3] SQLAlchemy Documentation, "SQLAlchemy ORM and Database Toolkit," Available: https://www.sqlalchemy.org/
[4] Google AI, "Gemini API Documentation," Available: https://ai.google.dev/docs
[5] P. Wozniak, "Optimization of Learning Using Spaced Repetition," SuperMemo, 1990. Available: https://www.supermemo.com
[6] Mozilla Developer Network, "Web Speech API (Speech Recognition and Synthesis)," Available: https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API
[7] J. D. Kelleher, B. Mac Namee, and A. D'Arcy, Fundamentals of Machine Learning for Predictive Data Analytics. MIT Press, 2015.
[8] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[9] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
[11] T. Mikolov et al., "Efficient estimation of word representations in vector space," in Proc. ICLR, 2013.
[12] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, 2017.
[13] D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd ed. Draft, 2023.
[14] F. Chollet, Deep Learning with Python. Manning Publications, 2017.
[15] C. D. Manning, P. Raghavan, and H. Schütze, Introduction to Information Retrieval. Cambridge University Press, 2008.