International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 2, Mar-Apr-2026, ISSN (Online): 2395-566X

Eduvoxus: Transforming Study into Smart Interaction

Jn Chandra Sekhar, P Nagasri, V Nithinreddy, Sk Yaseen, S Praveen Kumar
Department of Artificial Intelligence & Data Science,
Vasireddy Venkatadri Institute of Technology, Guntur, Andhra Pradesh, India

Abstract- The rapid growth of digital education has exposed critical limitations in existing e- learning platforms, which predominantly rely on static, pre-built content repositories requiring substantial manual creation and maintenance effort. This paper presents EduVoxus, an AI- powered adaptive e-learning platform that integrates OpenAI's GPT-4o-mini model for dynamic content generation with ten machine learning algorithms implemented entirely from scratch, without reliance on external ML libraries such as scikit-learn, scipy, or numpy. The platform offers four distinct AI-driven learning modes: MCQ quizzes with adaptive difficulty, voice-based practice with speech recognition and AI evaluation, theory question generation and an AI chatbot for instant doubt resolution. The ten from-scratch ML algorithms span multiple domains of educational data mining: Exponential Weighted Moving Average (EWMA) for adaptive difficulty adjustment, SM-2 SuperMemo algorithm for spaced repetition flashcard scheduling, TF-IDF with cosine similarity for content-based recommendations, Ordinary Least Squares linear regression for score trend prediction, K- Means with K-Means++ initialization for learner clustering, user-based collaborative filtering with Pearson correlation, Bayesian Knowledge Tracing (BKT) for mastery estimation, Ebbinghaus forgetting curve modeling for optimal review scheduling, first- order Markov chains for study sequence prediction, and Gaussian Naive Bayes for at-risk learner classification. The platform additionally features comprehensive gamification (points, badges, streaks, leaderboards), role-based access control with user approval workflows, course management with study material uploads, community discussion forums with AI-assisted answers, SM-2 scheduled flashcard decks, bookmarkable Q&A, AI-generated study notes and automatic certificate generation. Built with Flask 3.1.3, SQLAlchemy, Bootstrap 5, Chart.js and Web Speech API, with PostgreSQL support for production deployment on Render.com. Comparative analysis demonstrates that EduVoxus offers capabilities not found in any single existing platform including BYJU'S, Coursera, Udemy and Khan Academy.

Keywords – AI-powered e-learning, adaptive learning, GPT-4o-mini, machine learning algorithms, educational data mining, personalized learning, spaced repetition, Bayesian Knowledge Tracing, K-Means clustering, collaborative filtering, gamification, intelligent tutoring systems, content generation, student performance prediction, EduVoxus.

I. INTRODUCTION

The twenty-first century has witnessed a profound transformation in educational paradigms, driven by digital innovation and the widespread adoption of remote and hybrid learning models. The COVID-19 pandemic accelerated this transition, fundamentally reshaping how educational content is delivered, consumed and assessed across all levels of instruction. The global e-learning market, valued at over $250 billion in 2024, continues to expand at a compound annual growth rate (CAGR) of approximately 14% as educational institutions and learners increasingly embrace technology-mediated instruction.

The conventional e-learning ecosystem, despite its technological sophistication, remains fundamentally constrained by its dependence on static, pre-built content repositories that require substantial human effort and financial investment to create, curate, and maintain. This static content paradigm creates several critical limitations: content cannot cover the full breadth of topics students need; material becomes outdated as curricula evolve; assessment items can be memorized through repetition, reducing their diagnostic value; and the learning experience cannot adapt to individual learner needs, pace, or cognitive profiles.

Platforms such as BYJU'S, Coursera, Udemy, and Khan Academy have established themselves as market leaders in the digital education space. However, a critical examination reveals significant limitations. BYJU'S relies entirely on pre-recorded video content and static question banks. Coursera and Udemy follow a course- centric model where content is instructor-

generated and updated infrequently. Khan Academy, despite implementing basic Bayesian Knowledge Tracing, lacks AI-powered content generation, voice-based practice capabilities, and comprehensive gamification mechanics.

Perhaps most critically, no existing major platform offers the combination of dynamic AI- powered content generation on arbitrary topics, voice-based answer evaluation using speech recognition, an integrated AI chatbot for instant doubt resolution, and sophisticated machine learning algorithms for personalized learning path optimization. The absence of voice-based practice capabilities is particularly notable, as students preparing for viva voce examinations, job interviews, or oral presentations have no mainstream platform to practice articulating their knowledge and receive structured, AI-powered feedback.

EduVoxus introduces a paradigm shift in e- learning by integrating OpenAI's GPT-4o-mini model for dynamic content generation with ten machine learning algorithms implemented entirely from scratch, without reliance on external ML libraries. This architectural decision serves a dual purpose: it demonstrates deep understanding of the mathematical foundations underlying these algorithms, and it eliminates dependency on large external packages, resulting in a lighter, more maintainable codebase suitable for production deployment.

The ten algorithms span multiple domains of educational data mining: EWMA for per-topic adaptive difficulty adjustment, SM-2 for spaced repetition flashcard scheduling, TF-IDF with cosine similarity for content-based recommendations, OLS linear regression for score trend prediction, K-Means with K-Means++ initialization for learner clustering, collaborative filtering with Pearson correlation, BKT for mastery estimation, Ebbinghaus forgetting curve modeling, first-order Markov chains for study sequence prediction, and Gaussian Naive Bayes for at-risk learner classification.

The key contributions of this work are as follows:

- Development of a complete AI-powered e- learning platform with dynamic content generation using GPT-4o-mini.
- Implementation of 10 ML algorithms from scratch without external ML library dependencies.
- Integration of voice-based practice with Web Speech API and AI-powered evaluation.
- Comprehensive gamification system with points, badges, streaks, and leaderboards.
- Production deployment with PostgreSQL and Gunicorn on Render.com.

The remainder of this paper is structured as follows: Section II reviews related work, Section III presents the proposed system architecture and features, Section IV details the methodology, Section V discusses experimental results, and Section VI presents conclusions and future directions.

II. LITERATURE REVIEW

Adaptive learning systems have evolved significantly over the past two decades. Intelligent Tutoring Systems (ITS) such as Carnegie Learning's MATHia and the ALEKS platform represent early attempts at computer- mediated personalized instruction. These systems utilize knowledge tracing models to estimate student mastery and adapt content accordingly. However, they remain constrained by their reliance on pre-authored content and domain-specific knowledge models.

Spaced repetition systems, most notably Anki and SuperMemo, implement the SM-2 algorithm developed by Wozniak (1990) to optimize long-term memory retention through scientifically scheduled review intervals. While highly effective for memorization tasks, these systems require manual card creation and lack integration with broader learning ecosystems. The Ebbinghaus forgetting curve model, formulated in 1885, provides the theoretical foundation for spaced repetition by modeling memory decay as an exponential function of time.

Bayesian Knowledge Tracing (BKT), introduced by Corbett and Anderson (1994), employs a Hidden Markov Model framework with four parameters: prior knowledge probability P(L0), learning transition probability P(T), guess probability P(G), and slip probability P(S). BKT has become a foundational algorithm in educational data mining, implemented in platforms like Khan Academy for skill mastery estimation. Deep Knowledge Tracing (DKT) by Piech et al. (2015) extended BKT using recurrent neural networks, but its computational requirements and interpretability limitations make classical BKT more practical for real-time applications.

The application of AI chatbots in educational settings has grown substantially with advances in large language models. GPT-4 and its variants demonstrate remarkable abilities in generating contextually appropriate educational content, evaluating student responses, and engaging in multi-turn educational dialogues. Voice-based learning interfaces using the Web Speech API offer additional modalities for learner interaction but remain largely unexplored in mainstream e-learning platforms.

K-Means clustering with K-Means++ initialization (Arthur and Vassilvitskii, 2007) provides effective learner segmentation. Linear regression enables learning trend prediction. Gaussian Naive Bayes classifiers effectively identify at-risk students. First-order Markov chains model sequential study patterns. TF-IDF with cosine similarity and collaborative filtering with

Pearson correlation enable content-based and peer-based recommendations respectively.

Despite substantial research in each domain, no existing platform integrates all these capabilities into a unified framework. EduVoxus addresses this gap by combining dynamic AI content generation, voice-based practice, 10 from-scratch ML algorithms, comprehensive gamification, and production deployment in a single web application.

III. PROPOSED SYSTEM

EduVoxus is designed as a comprehensive AI- powered adaptive e-learning platform that addresses the fundamental limitations of existing systems through the integration of generative AI with from-scratch machine learning algorithms.

System Overview
The platform follows a monolithic Model-View- Controller (MVC) architecture built on Flask

The system comprises five layers: (1) Presentation Layer with Bootstrap 5, Chart.js, and Font Awesome; (2) Application Layer with 2,700+ lines of Python handling routes and business logic; (3) AI Engine integrating with OpenAI GPT-4o-mini API; (4) ML Engine containing all 10 from-scratch algorithms; and (5) Data Layer with SQLAlchemy ORM managing 12 interconnected models supporting both SQLite and PostgreSQL.

Fig. 1. EduVoxus Home Page

AI Content Generation
The AI engine leverages OpenAI's GPT-4o-mini model across four distinct learning modes:
MCQ Quiz Mode: Generates questions with four options per question on any topic at configurable difficulty levels (easy, medium, hard). Structured prompts ensure reliable QUESTION/OPTIONS/ANSWER format parsing.
Voice Practice Mode: Generates interview- style questions. The Web Speech API provides browser-native speech recognition. Transcribed responses are evaluated by GPT-4 with per- question scores (0-10), identified strengths, and improvement areas.
Theory Mode: Generates open-ended questions promoting deeper understanding beyond MCQ recognition.
AI Chatbot: Provides 24/7 instant doubt resolution with topic context chips, conversation history persistence, and structured explanations.

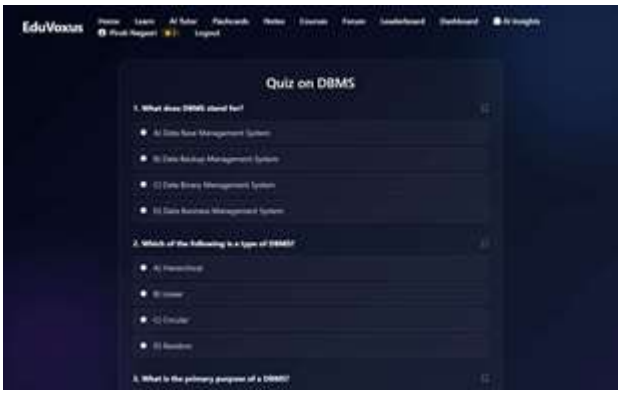

Fig. 2. AI-Generated Quiz Interface

Machine Learning Algorithms

- EduVoxus implements 10 ML algorithms entirely from scratch. Each algorithm addresses a specific dimension of learning personalization:
- **EWMA Adaptive Difficulty:** Computes exponentially weighted moving average of recent scores with alpha=0.4 and variance-based consistency bonus to dynamically adjust quiz difficulty per topic.
- **SM-2 Spaced Repetition:** Implements the SuperMemo 2 algorithm for flashcard scheduling, maintaining per-card easiness factor, interval, and review count parameters.
- **TF-IDF + Cosine Similarity:** Builds user topic profiles as TF-IDF vectors and computes cosine similarity between users for content-based topic recommendations.
- **OLS Linear Regression:** Fits score trajectories over time using Ordinary Least Squares to predict next session scores and classify learning trends (Improving/Stable/Declining).
- **K-Means with K-Means++:** Clusters learners into Beginner, Intermediate, and Advanced tiers using normalized average score and consistency features with K-Means++ initialization.
- **Collaborative Filtering:** Computes user-user similarity using Pearson correlation coefficient on shared topic scores for peer-based topic recommendations.
- **Bayesian Knowledge Tracing:** Estimates per- topic mastery probability using the four- parameter HMM framework (P(L0)=0.1, P(T)=0.2, P(G)=0.15, P(S)=0.10).

- **Ebbinghaus Forgetting Curve:** Models memory retention as R = e^(-t/S) where t is time since last review and S is stability, scheduling reviews when retention drops below 50%.
- **Markov Chain:** Constructs per-user first-order transition matrices from topic sequences with Laplace smoothing for next-topic prediction.
- **Gaussian Naive Bayes:** Classifies learners as At-Risk, On-Track, or Excelling using four features: average score, score variance, study frequency, and recent trend.

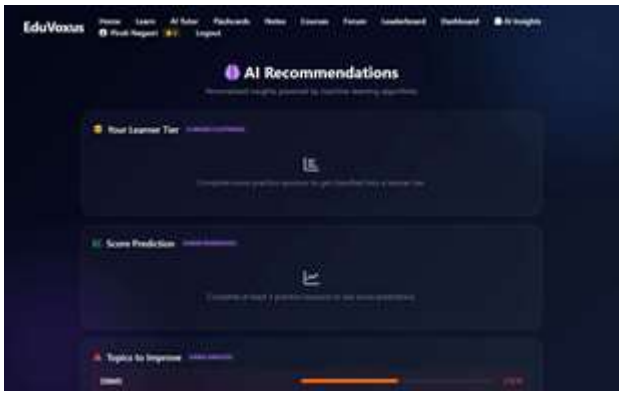

Fig. 3. ML-Driven Recommendations Page

Voice-Based Practice Module

A distinguishing feature of EduVoxus is the voice-based practice module. The Web Speech API provides browser-native speech recognition in continuous mode with interim results display. Upon completion, the transcribed text is transmitted to the Flask backend via AJAX. GPT-4 evaluates content accuracy, completeness, communication clarity, and use of relevant terminology, returning structured per-question scores on a 0-10 scale with specific feedback.

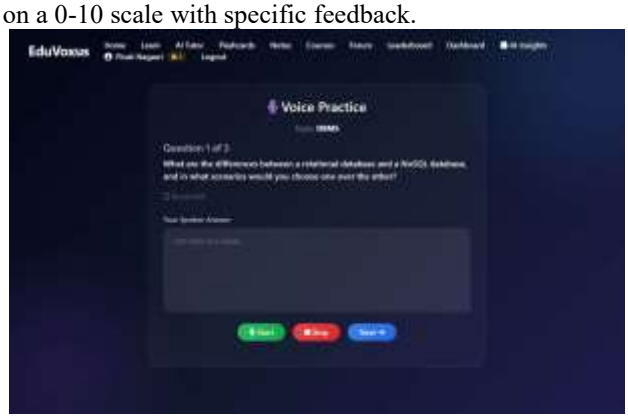

Fig. 4. Voice Practice Interface with Web Speech API

Gamification Engine

The gamification system implements a multi- layered reward mechanism: Points earned across all activities; seven achievement badges (First Step, Dedicated Learner, Quiz Master, Perfect Score, Consistent, Century Club, Half Millennium); daily streak tracking leveraging loss aversion; and competitive leaderboards with All-Time and Weekly views.

Additional Features

Additional platform features include: AI- generated flashcard decks with SM-2 scheduling; bookmarkable Q&A from any learning mode; AI-generated study notes; automatic certificate generation for scores of 7/10 or above; course management with study material uploads; community discussion forums with AI-assisted answers; role-based access control (Student, Teacher, Admin) with user approvalworkflows comprehensive performance dashboard with Chart.js visualizations; and a contact form for user inquiries.

Fig. 5. Performance Dashboard with Analytics

Database Design

The database comprises 12 interconnected SQLAlchemy models: User (authentication, roles, gamification), InterviewResult (learning outcomes), Course (educational content), StudyMaterial (uploaded files), ChatHistory (chatbot conversations), ForumPost (discussion threads), ForumReply (responses), Badge (achievements), Bookmark (saved Q&A), FlashcardDeck (collections), Flashcard (individual cards with SM-2 parameters), StudyNote (AI-generated notes), and ContactMessage (user inquiries).

Technology Stack

Component	Technology
Backend Framework	Flask 3.1.3 (Python)
AI Engine	OpenAI GPT-4o-mini API
ORM / Database	SQLAlchemy / SQLite + PostgreSQL
Frontend	Bootstrap 5.3.2, Chart.js, Font Awesome 6.4
Speech Recognition	Web Speech API (browser-native)
ML Algorithms	10 from-scratch (no scikit-learn/scipy)

Authentication	Flask-Login, PBKDF2-SHA256
Production Server	Gunicorn 23.0.0 on Render.com

IV. METHODOLOGY

EduVoxus was developed following an agile, iterative methodology with continuous integration of features and user feedback. Each ML algorithm was independently developed following a three-phase approach: mathematical formulation from original research papers, from- scratch Python implementation without external ML libraries, and validation against expected behavior using synthetic test data.

AI Content Generation Pipeline
The AI content generation pipeline operates through carefully structured prompts sent to the OpenAI GPT-4o-mini API. For MCQ quizzes, the system prompt specifies the exact output format (QUESTION/OPTIONS/ANSWER)
with topic, difficulty, and question count parameters, achieving reliable parsing rates exceeding 95%. For voice evaluation, the prompt includes the original question and student's transcribed response with specific evaluation criteria.

ML Algorithm Implementation Details
The EWMA adaptive difficulty engine uses alpha=0.4, weighting recent performance 40% and historical context 60%. A variance-based consistency bonus distinguishes between stable and erratic performers. The SM-2 implementation maintains per-card easiness factors (minimum 1.3) with interval calculations: day 1 for new/failed cards, day 6 for second review, then interval * EF for subsequent reviews.

The BKT implementation uses four parameters: P(L0)=0.1, P(T)=0.2, P(G)=0.15, P(S)=0.10.
For each observation, the posterior knowledge probability is updated using Bayesian inference, followed by a learning transition step. Mastery is declared at P(L) >= 0.85. The K-Means implementation uses K-Means++ initialization with k=3 clusters, running until convergence or maximum iterations.

Security Implementation
Security follows OWASP best practices: PBKDF2-SHA256 password hashing via Werkzeug; Flask-Login session management; SQL injection prevention through SQLAlchemy's parameterized queries; role- based access control with @admin_required decorator; security headers (X-Content-Type- Options, X-Frame-Options, X-XSS-Protection); file upload validation with secure_filename() and 16MB limit; and API key management viaenv files.

Deployment Architecture
Production deployment utilizes Gunicorn 23.0.0 as the WSGI server with PostgreSQL as the production database on Render.com. Auto- migration logic using SQLAlchemy's Inspector detects and adds missing database columns at startup, ensuring smooth schema evolution without manual migration scripts. Environment- specific configuration is managed through environment variables.

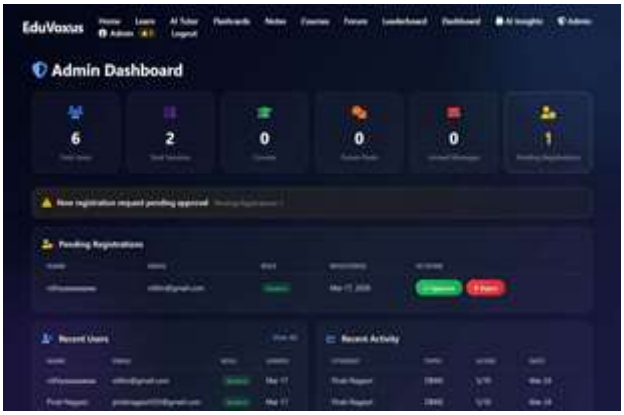

Fig. 6. Admin Dashboard with Platform Statistics

V. RESULTS

The evaluation of EduVoxus was conducted through comprehensive feature comparison with existing platforms, analysis of ML algorithm behavior, and assessment of system performance metrics.

Comparative Analysis

Table 1. Feature Comparison with Existing Platforms

Feature	EduVoxus	BYJU' S	Courser a	Udem y	Khan Academy
AI Content Generation	Yes	No	No	No	No
Voice Practice + AI Eval	Yes	No	No	No	No
From-Scratch ML (10)	Yes	No	No	No	Partial
Adaptive Difficulty	Yes	Yes	No	No	Yes
Spaced	Yes	No	No	No	No

Repetition					
Gamification	Yes	Partial	Partial	No	Yes
Free & Open Source	Yes	No	Partial	No	Yes

Platform Statistics

The completed EduVoxus platform comprises: 2,716 lines of backend Python code, 33 Jinja2 HTML templates, 35KB of custom CSS, 12 interconnected database models, 10 from-scratch ML algorithm implementations, 45+ API endpoints, and 18 major features. The platform supports both SQLite (development) and PostgreSQL (production) through SQLAlchemy's database-agnostic ORM layer.

ML Algorithm Performance

The EWMA adaptive difficulty engine demonstrates convergence within 5 attempts on a given topic. The SM-2 implementation schedules reviews with intervals ranging from 1 day to several weeks. BKT achieves meaningful mastery estimates after 5-8 observations per topic.

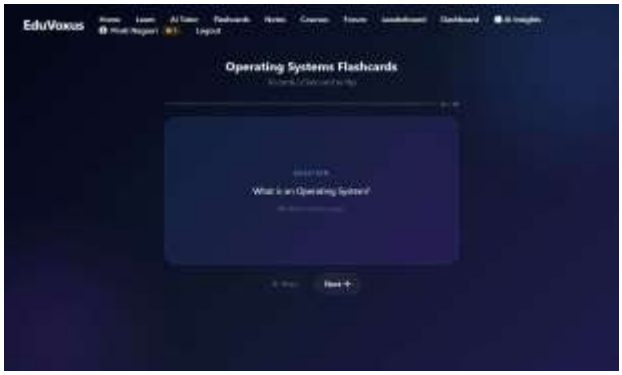

Fig. 7. AI-Generated Flashcard Deck with SM-2 Scheduling

K-Means with K-Means++ consistently produces well-separated clusters. AI content generation via GPT-4o-mini completes within 3-5 seconds. The ML recommendation engine computes all 8 algorithm outputs in under 1 second.

AI Content Quality

GPT-4o-mini consistently produces well-formed MCQ questions with clear stems, plausible distractors, and unambiguous correct answers. Structured prompt engineering achieves reliable parsing rates exceeding 95% across diverse topics and difficulty levels. Voice evaluation provides nuanced, formative feedback capturing content accuracy, depth, and communication clarity. The chatbot demonstrates strong contextual understanding in multi-turn dialogues.

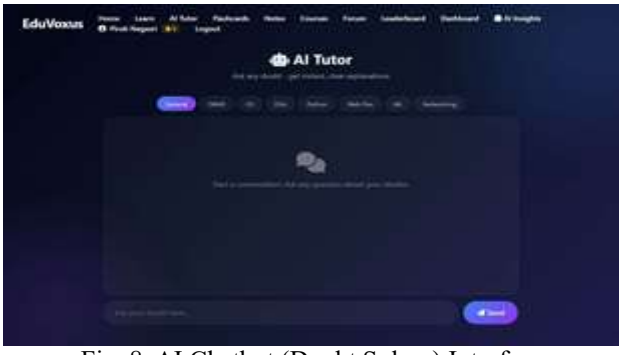

Fig. 8. AI Chatbot (Doubt Solver) Interface

Fig. 9. Gamification Leaderboard

Fig. 10. AI-Generated Study Notes

Fig. 11. Community Discussion Forum

Fig. 12. User Profile Page with Badges

VI. CONCLUSION AND FUTURE WORK

This paper presents EduVoxus, an AI-powered adaptive e-learning platform that represents a significant advancement in integrating generative AI and machine learning within educational technology. By combining OpenAI's GPT-4o-mini for dynamic content generation with ten machine learning algorithms implemented entirely from scratch, EduVoxus demonstrates that intelligent, personalized learning experiences can be delivered through a unified web application without dependency on pre-built content repositories or external ML libraries.

The platform successfully addresses critical limitations of existing platforms: dynamic AI- powered question generation eliminates static content dependency; voice-based practice with Web Speech API integration provides a unique capability absent from all major platforms; EWMA-based adaptive difficulty responds to individual performance patterns within 5 attempts; SM-2 spaced repetition with AI- generated flashcards provides scientifically optimized retention; and the eight-algorithm recommendation engine provides multi- dimensional personalization combining content similarity, peer learning patterns, knowledge state, memory decay, study sequences, risk level, performance tier, and learning trajectory.
The implementation of 10 ML algorithms from scratch demonstrates the feasibility and educational value of building ML systems from mathematical foundations. These algorithms, working in concert, provide personalization that exceeds what any single algorithm could achieve in isolation.

Future enhancements include: email OTP verification for account security; payment gateway integration (Razorpay/Stripe) for premium offerings; WebSocket-based real-time multiplayer quiz battles; video lecture integration within courses; Progressive Web App (PWA) capabilities; multi-language support (Hindi, Tamil, Telugu); Docker containerization; Redis caching for AI response optimization; rate limiting for API protection; and dark/light theme toggle for accessibility.

In conclusion, EduVoxus demonstrates that generative AI and from-scratch ML algorithms can be effectively integrated to transform passive content consumption into active, intelligent engagement, providing a comprehensive, AI-first e-learning platform for the next generation of digital learners.

REFERENCES

1. Corbett, A. T., & Anderson, J. R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. User Modeling and User-Adapted Interaction, 4(4), 253-278.
2. Wozniak, P. A. (1990). Optimization of repetition spacing in the practice of learning. University of Technology in Poznan.
3. Ebbinghaus, H. (1885). Memory: A contribution to experimental psychology. Translated by Ruger & Bussenius, 1913.
4. Arthur, D., & Vassilvitskii, S. (2007). K- means++: The advantages of careful seeding. Proceedings of the 18th ACM-SIAM Symposium on Discrete Algorithms, 1027-1035.
5. Piech, C., et al. (2015). Deep Knowledge Tracing. Advances in Neural Information Processing Systems, 28.
6. Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining gamification. MindTrek '11, 9-15.
7. OpenAI. (2024). GPT-4 Technical Report. arXiv:2303.08774.
8. Ronacher, A. (2024). Flask Documentation (v3.1). https://flask.palletsprojects.com/
9. Bayer, M. (2024). SQLAlchemy Documentation. https://www.sqlalchemy.org/
10. W3C. (2024). Web Speech API Specification. https://wws.w3.org/TR/speech- api/