ABACUS – AI Powered Learning Prompter

Prof. P. G. Salve [*1] , Ms. Ingale Manasvi Sunil [*2], Ms. Rajguru Aditi Mahesh [*3], Ms. Vidhate Deepali Pramod [*4], Ms. Shewale Hiral Dinesh [*5]

*1 Prof, Department of Computer Technology, K. K. Wagh Polytechnic, Nashik, Maharashtra, India.

*2,3,4,5 Student, Department of Computer Technology, K. K. Wagh Polytechnic, Nashik, Maharashtra, India.

Abstract

The **ABACUS – AI Powered Learning Prompter** is a web-based educational application designed to modernize traditional abacus training by providing an interactive learning environment enhanced with text-to-speech capabilities. The system caters to both Admin and Student roles, enabling teachers to manage arithmetic question banks and students to practice and evaluate their performance. A key feature of the system is voice integration, which reads out arithmetic expressions aloud to support auditory learning and make the software accessible to learners of varying abilities.Built using a React.js frontend, PHP backend, and MySQL database, this application facilitates smooth interactions, real-time feedback, and robust data management. The voice support component enhances engagement and learning effectiveness by allowing students to hear problems as well as see them on screen. The system aims to make abacus practice intuitive, scalable, and suitable for schools, coaching centers, or independent learners.

INTRODUCTION

The Abacus is one of the oldest and most effective tools used for teaching arithmetic and mental calculation. Traditionally, learners use a physical frame with sliding beads to perform mathematical operations such as addition, subtraction, multiplication, and division. Though simple in design, the abacus develops strong mental arithmetic skills, boosts concentration, and enhances cognitive abilities such as memory and analytical thinking. It continues to be used in education worldwide due to its effectiveness in strengthening numerical intuition and brain development.

Despite its longstanding value, traditional abacus training often relies on in-person instruction, static problem sets, and manual evaluation. These methods can be limiting in terms of scalability, interactivity, and personalized learning support for students of varying abilities. Additionally, modern learners increasingly expect digital solutions that adapt to individual pace, provide instant feedback, and support multiple sensory modes of learning.

To address these limitations, the ABACUS – AI Powered Learning Prompter project introduces an interactive, web-based learning platform that integrates modern software technologies with abacus methodology. The system enhances traditional practice with structured digital content, instant feedback, and voice integration that reads arithmetic expressions aloud. This not only supports visual learning but also aids auditory learners and learners with accessibility needs. The platform features separate access panels for administrators (teachers) and students, enabling efficient question management and personalized practice paths. With the use of web technologies such as React.js for frontend, PHP/MySQL for backend, and Text-to-Speech (TTS) for voice support, the system aims to modernize abacus training in an engaging and scalable format.

1. PROBLEM STATEMENT

Traditional abacus training, while effective for developing basic arithmetic and cognitive skills, suffers from several limitations that hinder its broader adoption in modern education. First, learning abacus with physical tools is time-consuming and requires consistent practice, which many students find tedious and difficult to maintain over long periods. Second, the abacus is limited to basic arithmetic operations and lacks interactive features that support higher engagement, real-time feedback, and adaptive learning. Additionally, conventional methods lack accessibility and scalability, as students must attend physical classes or rely on static worksheets, making it hard for learners in remote areas or those with accessibility needs to benefit equally. Existing training also does not provide auditory support to assist learners who benefit from multisensory instruction, and educators have limited tools for managing and tracking student progress seamlessly. These issues highlight the need for a digital learning platform that integrates interactive practice, automated feedback, multisensory support, and efficient content management to modernize abacus education and make it more engaging, accessible, and effective.

2. LITERATURE SURVEY

The literature review explores research on digital abacus learning tools, e-learning environments, and technology integration in education — all of which provide background and justification for developing the ABACUS – AI Powered Learning Prompter system.

Sr.	Title / Author	Date / Venue	Key Relevance	Key Disadvantages / Challenges
1.	**Abacus: Precise Side-Channel Analysis** Q. Bao, et al.	**May 2021** ICSE (IEEE/ACM)	Quantifies **leaked bits** in binary programs. Helps prioritize severe vulnerabilities in libraries like OpenSSL.	Single-trace focus; may miss unexplored paths. Computationally expensive for large binaries.
2.	**Abacus: A Tool for Precise Analysis** Q. Bao, et al.	**2021** Tool Demo (ICSE)	Practical **workflow and CLI** guide. Supports both stripped and non-stripped binaries for testing.	High RAM/CPU demands. Requires manual configuration of secrets/paths.
3.	**Review of NLP in Language Learning** Jiale Peng	**2024** ICCDS	Categorizes NLP into **Reading, Writing, Oral, and Personalization.** Discusses the transition from CALL to **I-CALL**.	**Accuracy:** Struggles with idioms/slang. **Privacy:** Data collection risks. **Cultural Adaptability.**
4.	**AI-Enabled Adaptive Learning Systems: Systematic Mapping** T. Kabudi, et al.	**2021** Comp. & Ed: AI	Maps **147 studies** (2014-2020). Identifies **Bayesian Networks** and **Neural Networks** as top techniques for personalization.	**Implementation Gap:** Most systems are in experimental phases and rarely used in real-world "ordinary" courses.

3. METHODOLOGY

The methodology of the **ABACUS – AI Powered Learning Prompter** outlines the systematic approach followed in designing and developing the web-based abacus learning system with voice integration. This section describes how the project phases were planned, executed, and integrated to deliver an interactive, user-friendly educational platform.

a) Requirement Analysis and System Planning

Initially, the functional and non-functional requirements were gathered through discussions with educators and potential users. The core features were defined, including separate Admin and Student modules, question management, practice and test modules, and Text-to-Speech (TTS) voice support. System planning involved creating use cases, identifying data flows, and defining interfaces between components.

b) System Design

The overall system architecture was designed based on the Model-View-Controller (MVC) pattern to ensure scalability and maintainability. The frontend (React.js) was separated from the backend (PHP) with clearly defined API endpoints. Database schemas for users, questions, and test results were finalized to support efficient data handling.

c) Development and Integration

The development was carried out in modular phases:

i. **Frontend Development:**

React.js was used to build the user interface with dedicated components for login, practice sessions, tests, and result display. Navigation was managed using React Router, and state was handled using React Hooks to manage question sets and user sessions effectively.

ii. **Backend Development:**

A RESTful API was implemented in PHP, handling secure user authentication, question CRUD operations, test generation, and result submission. A MySQL database was configured for storing users, questions, tests, and performance data.

iii. **Voice Integration:**

Voice functionality was implemented using the Web Speech API’s speechSynthesis interface. During practice and tests, arithmetic expressions were processed into voice output, allowing students to hear each question aloud.

d) Testing and Validation

Once modules were developed, systematic testing was conducted to validate functionality. Test cases were prepared for login flows, CRUD operations, voice output, and test submission to ensure correct behavior under multiple scenarios.

e) Deployment & Feedback Integration

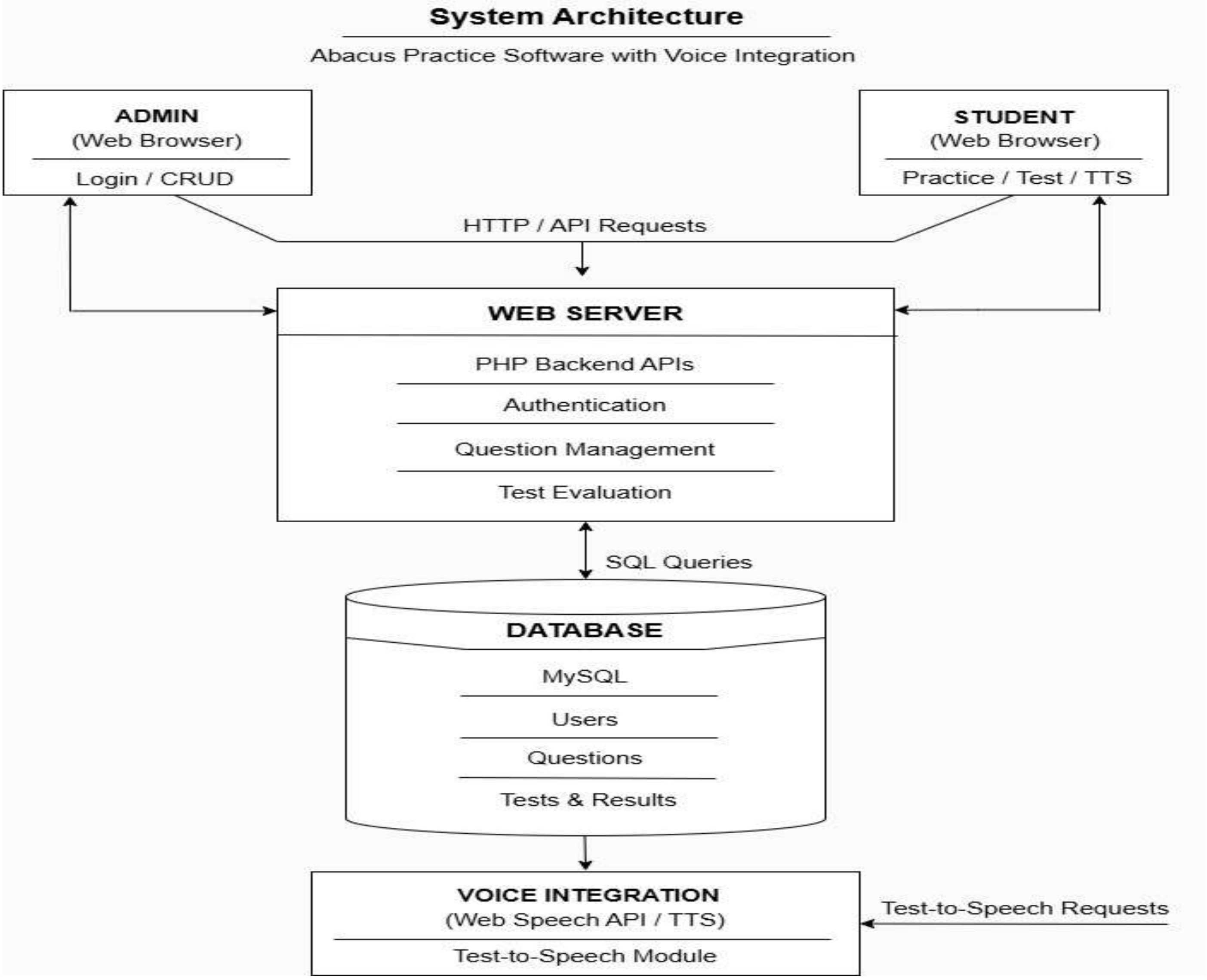
After successful testing, the application was deployed on a web server. User feedback was collected to refine UI elements and improve voice feature responsiveness. Iterative fixes were applied, ensuring a stable and effective learning environment.

4. SYSTEM ARCHITECTURE AND SOFTWARE DESIGN OF ABACUS – AI POWERED LEARNING PROMPTER

The **System Architecture and Software Design** of the *ABACUS – AI Powered Learning Prompter* describes how all the technical parts of the application work together to deliver an interactive and voice-assisted learning experience. It defines the **structure, interactions, components, and design patterns** used to build a scalable, secure, and user-friendly learning platform.

a) System Architecture Overview

The architecture of the *ABACUS – AI Powered Learning Prompter* follows a **multi-tiered web application model** that separates user interaction, application logic, and data management. This layered structure ensures the application is maintainable, scalable, and easy to enhance in the future:



1. Admin (Web Browser)

- The Admin accesses the system through a web browser.
- Admin performs **Login and CRUD operations** (Create, Read, Update, Delete) on arithmetic questions.
- Questions are categorized based on difficulty levels (Easy, Medium, Hard).
- Admin requests are sent to the web server using HTTP/API calls.

2. Student (Web Browser)

- The Student accesses the system via a web browser.
- Students can **practice questions, take tests, and use Text-to-Speech (TTS)** functionality.
- Student actions such as answering questions or starting tests are sent to the server through HTTP/API requests.

3. Web Server (PHP Backend APIs)

- The web server acts as the core processing unit of the system.
- It handles:
  - Authentication of Admin and Students
  - Question Management
  - Test Evaluation and Result Calculation
- The server processes incoming requests and interacts with the database using SQL queries.

4. Database (MySQL)

- The MySQL database stores all system data securely.
- It contains:
  - User information (Admin & Student)
  - Arithmetic questions
  - Test details and results
- The database sends required data back to the web server for processing.

5. Voice Integration (Web Speech API / TTS)

- This module is used mainly by the Student.
- Arithmetic questions are converted into spoken words using Text-to-Speech (TTS**)**.
- Example: “7 + 2 =” is spoken as “Seven plus two equals”.
- The voice request is triggered from the student interface.

6. Audio Output

- The generated speech is delivered as audio output to the student.
- This improves accessibility and supports auditory learning.

b) Architectural Design Patterns

To make the system organized and easy to manage, the application uses standard architectural patterns:

Model-View-Controller (MVC) Pattern

- The MVC pattern divides the application’s logic into distinct components:
  - **Model:** Manages data and database interaction.
  - **View:** Handles how information is presented to the user.
  - **Controller:** Acts as a bridge, interpreting user actions and coordinating responses between Model and View.

This separation allows developers to edit one part of the system (e.g., user interface) without affecting other parts (like data handling).

c) Component Interactions

The system can be visualized as several interacting components:

1. **Frontend Components:**
   - Login/Registration Pages
   - Admin Dashboard (Manage Questions)
   - Practice Interface (With Voice Button)
   - Test Interface
   - Result & Performance Dashboard
2. **APIs:**
   - Authentication API
   - Question Management API
   - Test Processing API
   - Voice Prompt API
3. **Database Entities:**

- ○ users
- ○ questions
- ○ tests
- ○ results

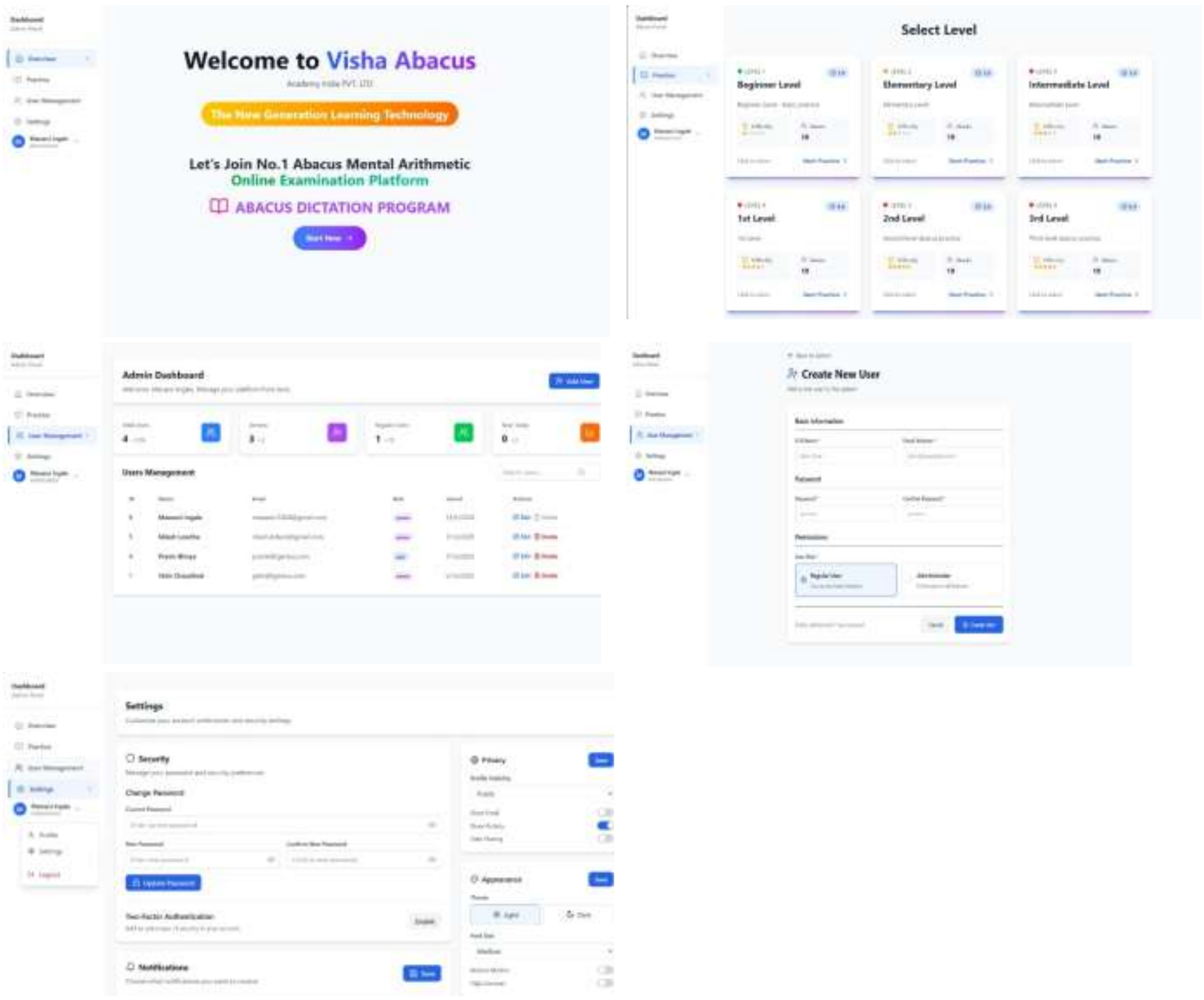
Each frontend component sends API requests to the backend, which in turn retrieves or updates data from the database. For example, when a student clicks "Listen", a request triggers a Text-to-Speech process that converts arithmetic expressions into voice output.

d) Security and Scalability Considerations

To maintain data protection and ensure seamless operation:

- **Secure Authentication:** Passwords are stored using hash encryption, and sessions are managed securely.
- **Role-Based Access:** Admin and Student roles have different permissions to prevent unauthorized actions.
- **Scalable Design:** REST-based APIs and modular frontend components allow future expansion (like mobile apps or advanced AI features).

5. Image



6. SOFTWARE DESCRIPTION

The ABACUS – AI Powered Learning Prompter is a **web-based educational application** designed to enhance abacus practice through interactive features and voice-assisted learning. It combines modern web technologies to deliver a dynamic, secure, and user-friendly interface that supports both **students** and **administrators** in the learning process.

a) Front-End Software Components

- **React.js** – Builds dynamic and responsive user interfaces.
- **HTML5** – Defines structure of web pages.
- **CSS3** – Styles layout and appearance of UI.
- **JavaScript (ES6+)** – Controls client-side logic and interactions.
- **Web Speech API** – Provides Text-to-Speech for voice prompts during practice/tests.

b) Back-End Software Components

- **PHP (Hypertext Preprocessor)** – Handles server-side logic and APIs.
- **RESTful API Endpoints** – Enable communication between frontend and backend.
- **Session Management & Security** – Ensures user authentication and protection.

c) Database Software

MySQL Database –

- A relational database management system used to store structured data for users, questions, tests, and results.
- Supports SQL for queries and efficient data retrieval.
- Offers high performance, reliability, and scalability for web applications.
- Ensures data integrity and secure storage through ACID compliance and authentication features.

Core Tables:

a. users – Stores student/admin credentials and roles.
b. questions – Stores abacus practice and test questions.
c. tests – Stores metadata for each test session.
d. results – Records students' scores and analytics.

d) Development Tools

- **Visual Studio Code** – Code editor for frontend/backend development.
- **XAMPP/WAMP Server** – Local development environment (Apache + PHP + MySQL).
- **Web Browsers:**

- Google Chrome
- Mozilla Firefox
- Microsoft Edge

- **API Test Tools:**

- Postman / Insomnia for backend API testing.

e) Additional Software Features

- **REST API Structure:**

- Supports GET, POST, PUT, DELETE methods for data operations.

○ **Responsive UI:**

• Designed to work on desktops, tablets, and mobile browsers.

○ **Voice Integration Module:**

• Enables audible reading of arithmetic expressions to assist learning.

7. APPLICATION

- Supplementary instructor aid for classroom & remote learning.
- Corporate training micro-modules and on-the-job refresher prompts.
- Academic institutions, coaching centers, corporate microlearning, remote education initiatives.

8. ADVANTAGES

a) **Wide Accessibility:** Students can access learning materials anytime and from anywhere with an internet connection, removing geographical limitations.

b) **Flexible Learning:** Learners can practice at their own pace according to their schedule, leading to better retention.

c) **Cost-Effective:** Reduces expenses related to printed materials, physical classrooms, and instructor travel.

d) **Interactive and Engaging:** Tools like instant feedback, voice integration (TTS), quizzes and interactive interfaces enhance student engagement and motivation.

e) **Scalable Content Management:** Admins can update, add or delete question sets efficiently without manual paperwork.

f) **Personalized Learning Experience:** Students can revise content, repeat practice tests and focus on areas they find difficult.

9. DISADVANTAGES

a) **Limited Face-to-Face Interaction:** Online formats may reduce direct interaction between teachers and learners, which can impact communication and collaboration skills.

b) **Dependence on Internet & Technology:** Users need a stable internet connection and suitable devices; poor connectivity can disrupt learning.

c) **Requires Self-Discipline:** Without structured classroom environments, students may struggle to stay motivated and maintain consistent study routines.

10. CONCLUSION

The ABACUS – AI Powered Learning Prompter effectively combines traditional abacus training with modern web and voice technologies to create an engaging and accessible learning environment. The platform enhances accessibility, flexibility, and personalized practice, while the integrated voice synthesis supports auditory learners and improves usability. However, it also shares common challenges of e-learning systems, including dependency on technology, need for strong internet access, and reduced face-to-face interaction. Overall, when implemented with supportive instructional design and reliable infrastructure, the system can significantly improve mental arithmetic learning outcomes and broaden access to abacus education.

11. REFERENCES

a) **Main Abacus Research Paper** Bao, Q., Wang, Z., Li, X., Larus, J. R., & Wu, D. (2021). Abacus: Precise side-channel analysis. In *Proceedings of the 2021 IEEE/ACM 43rd International Conference on Software Engineering (ICSE)* (pp. 797–809). IEEE. https://doi.org/10.1109/ICSE43902.2021.00078

b) **Abacus Tool Companion Paper** Bao, Q., Wang, Z., Larus, J. R., & Wu, D. (2021). Abacus: A tool for precise side-channel analysis. In *2021 IEEE/ACM 43rd International Conference on Software Engineering: Companion Proceedings (ICSE-Companion)* (pp. 238–239). IEEE. https://doi.org/10.1109/ICSE-Companion52605.2021.00110

c) **NLP in Language Learning Review** Peng, J. (2024). A comprehensive review of the application of NLP technology in language learning. *Applied and Computational Engineering, 92*(1), 163–168. https://doi.org/10.54254/2755-2721/92/20241735

d) **AI Adaptive Learning Systematic Mapping** Kabudi, T., Pappas, I., & Olsen, D. H. (2021). AI-enabled adaptive learning systems: A systematic mapping of the literature. *Computers and Education: Artificial Intelligence, 2*, 100017. https://doi.org/10.1016/j.caeai.2021.100017