



Vocora: AI-Powered Language Reinforcement Platform

2025 Final Project Report

CSCI 4390-02

Instructor: Dr. Nurbekov

Prepared by:

Andrea Garza

Teresa Garza

Mariana Resendez

Perla Lopez

Spring 2025

Prepared for:

University of Texas Rio Grande Valley

College of Engineering and Computer Science

ABSTRACT

Vocora is a full-stack web application designed to assist learners reinforce their language skills through AI-generated stories, synthesized audio, and tailored imagery. Developed using Next.js/React with TypeScript and styled via Tailwind TS, our web application utilizes Supabase for user authentication, row-level security, and data storage. OpenAI's o1-mini-2024-09-12 model facilitates story-generation along word translations, part-of-speech, and definitions in context along with its TTS-1 model which delivers audio narration of each story; meanwhile, Fireworks AI's stable-diffusion-xl-1024-v1-0 generates high-quality illustrations. Conversational chat and writing evaluation features, both powered by Google Gemini-2.0-flash API model, offers real-time practice. This report outlines our system architecture, emphasizes core features, and details the contributions of each team member. Key outcomes include seamless multilingual support, a story-hash caching system to reduce API costs, and RLS policies to maintain data privacy. Future developments will focus on voice recognition, progress tracking, along with adding more languages.

INTRODUCTION

Multiple language-learning resources are targeted at beginners, but intermediate learners require engaging, context-aware practice to maintain fluency. Our web application Vocora addresses this gap by generating personalized stories that utilizes user-selected vocabulary, providing contextual definitions and visuals to enhance retention. Our goal was to develop a platform where users can log in, select words, read and hear narratives, view tailored illustrations, and interact with our AI companion “Vocora.”

PROJECT OVERVIEW

- Frontend:
 - Next.js
 - React
 - TypeScript
 - Tailwind TS
- Backend:
 - Supabase (PostgreSQL, Authorization, Row-Level Security policies)
- AI Services:
 - OpenAI o1-mini-2024-09-12
 - Story generation, contextual definitions
 - OpenAI TTS-1
 - Audio generation
 - Fireworks AI stable-diffusion-xl-1024-v1-0
 - Image generation
 - Google Gemini-2.0 flash
 - Chat functionality, interactive writing practice
- Core Features
 - Multilingual UI and Personalization
 - Users can select their native and practice languages; both UI and word definitions adapt seamlessly.
 - Story Generation and Caching
 - When a word is selected Vocora generates a narrative, labels each word with its part of speech, translation, definition, and saves it using a story-hash system to prevent duplicate API calls to OpenAI.
 - Interactive Practice
 - A chat system preserves conversation history and provides relevant responses to reinforce additional learning.
 - Security and Data Protection
 - RLS policies ensure authenticated users can only access their own vocabulary and stories, with a shared user (Vocora) for defaulted words.

INDIVIDUAL CONTRIBUTIONS

Perla Lopez

- Project Setup
 - Opted for TypeScript due to its strong typing benefits and enhanced developer experience.
 - Initialized the Next.js project with TypeScript configuration.
- UI/UX Design
 - Designed the web application's aesthetic, establishing a cohesive visual style.
 - Implemented both light and dark mode toggles, ensuring accessible color contrast and smooth transitions between themes.
- Dashboard Redesign
 - Redesigned the original dashboard layout to enhance usability, reorganizing vocabulary lists, story and writing access, along chat controls into user-friendly sections.
 - Implemented responsive design enhancements for various screen sizes, improving the user experience on both desktop and mobile platforms.

Teresa Garza

- Translation Implementation
 - Led beginning-to-end translation of the UI by consolidating all text into a single translation file, removing the previous per-language routes approach.
 - Organized translations into separate sections (en, es, zh) , facilitating dynamic access based on each user's preferred language.
 - Integrated translation hooks throughout all components, ensuring seamless language switching and consistent terminology throughout the site.
- Writing Practice Tutor
 - Implemented logic for the grammar AI tutor using Google Gemini's model, evaluating user input and delivering corrections along with explanations in the targeted practice language.
 - Integrated UI components for grammar feedback , which highlights mistakes inline and provide suggestions within context.
- AI Chatbot
 - Developed a conversational chatbot utilizing the Google Gemini model, enabling users to participate in prolonged dialogues when they feel prepared to take on full conversations.
 - Preserved chat history for contextual understanding, and designed UI elements for message threading and input controls.

Andrea Garza

- Routing and Authentication
 - Implemented Next.js route handlers and integrated Supabase Auth with regular signup and Google OAuth.
 - Solved prerendering issues to ensure pages load in the user's preferred language immediately after login.
- Database Design and Security
 - Created tables for vocab_words, vocab_lists, cached_definitions, user_stories, and user_preferences.
 - Wrote and tested RLS policies for all CRUD operations, implementing specific logic to assign default words to new users from a shared user ID.
 - Integrated user_stories to store generated stories in Supabase, allowing users to retrieve each story's metadata and content. Each story was stored separately depending on the user's practice language.
- Story Generation
 - Implemented logic for story generation using OpenAI's model, mapping user-selected vocabulary into each narrative.
 - Managed front-end state to highlight user-selected vocabulary within the stories.
- Image Generation
 - Connected Fireworks AI's endpoint to generate tailored story illustrations on demand.
- Code Refactoring
 - Cleaned and organized the codebase by removing unused code and modularizing components for better maintainability.
 - Optimized algorithms and data-fetching processes to improve runtime efficiency and minimize loading times.

Mariana Resendez

- Text To Audio Generation
 - Implemented audio playback functionality using OpenAI's "TTS-1" Text-to-Speech model to convert generated story text into audio.
 - Ensured the audio player was rendered only if the story generation was successful/TTS output was valid.
 - Enable users to listen to generated stories for pronunciation practice and improved language comprehension.
- Figma Design
 - Created initial wireframes and mockups for essential pages like the story-generation page.
 - Designed and illustrated logo and mascot character "Vocora."
- Website SiteMap
 - Drafted initial structure for site navigation to ensure navigation between key pages (log-in, story-generation)
 - Edited our updated UI through removing unused features and ensuring accurate multilingual translations for updated landing page's feature summary tabs.
- Chat-Box UI
 - Completed implementation of chat box pop-up interface resembling support chats, where users can interact with Vocora for vocabulary practice.
 - Illustrated and integrated new pose of mascot to enhance interactivity within chat feature.

CONCLUSION

Our platform Vocora works to help facilitate a user's language journey through vocabulary reinforcement. Using AI-generated stories paired with audio and image elements, users are able to engage with vocabulary terms in their respective contexts and save these words and stories for review or further practice. Our chat-box feature allows users to practice with our AI powered assistant "Vocora" for conversation practice that feels natural to the user. With this approach, users are able to learn more efficiently and enjoyably, and are able access this content at any level. Our team is proud of its current progress, and hopes to further improve this application.

Moving forward, we plan to:

1. Implement voice recognition/practice
 - Allows users to provide audio input and receive feedback on pronunciation
2. Integrate progress tracking
 - Enable users to visually monitor how often they correctly identify and practice each vocabulary term.
3. Expand language support
 - Include languages such as French, German, and Arabic through translation files.
 - Research and integrate AI models specialized for each practice language to enhance real-time grammar feedback.
4. Optimize performance
 - Support large vocabulary lists by introducing search filters, grouping terms into categories, etc.

REFERENCES

1. Supabase. (2024). *Row Level Security | Supabase Docs*. Retrieved from:
<https://supabase.com/docs/guides/database/postgres/row-level-security>
2. Supabase. (2023). *Auth | Supabase Docs*. Retrieved from:
<https://supabase.com/docs/guides/auth>
3. OpenAI. *o1-mini model-Open AI API | OpenAI Platform*. Retrieved from:
<https://platform.openai.com/docs/models/o1-mini>
4. OpenAI. *TTS-1 model-Open AI API | OpenAI Platform*. Retrieved from:
<https://platform.openai.com/docs/models/tts-1>
5. Fireworks. (2025). *Stable Diffusion XL | Stability AI*. Retrieved from:
<https://fireworks.ai/models/fireworks/stable-diffusion-xl-1024-v1-0>
6. Google. (2025). *Gemini API Docs and Reference | Google AI for Developers*. Retrieved from: <https://ai.google.dev/gemini-api/docs>