

**Introduction**

The increasing demand for mobile-first educational tools has created opportunities to design applications that blend technology with cultural relevance. Mzansi Lingo is positioned to address this need by offering an inclusive, interactive platform for learning South Africa’s diverse languages. The system integrates gamification, structured lessons, and speech-centred learning to make language acquisition engaging and effective. A key component of the design is the conversational feature, which enables learners to practise their speaking and comprehension skills in real time. This element ensures that users are not only exposed to vocabulary and grammar but are also encouraged to apply their knowledge in a practical, feedback-driven environment (CivicScience, 2023).

**Overview of the Application**

**App Name:** Mzansi Lingo

**App logo:** The logo features a friendly cartoon green rhinoceros head on a bright yellow background. The rhino has large round eyes with sparkles, a cheerful smile, white horns, and is drawn in a modern flat illustration style with bold black outlines. The playful design makes language learning feel approachable and fun, while the rhino connects to South African identity, perfectly suiting the "Mzansi Lingo" brand.

**Brief Description:** Mzansi Lingo is a mobile-first, interactive language-learning application that brings the joy of learning to South Africa’s diverse cultural and linguistic heritage. It offers courses in seven official South African languages: isiZulu, isiXhosa, Sesotho, Setswana, Sepedi, Afrikaans, and English. The app combines gamified learning, cultural immersion, and practical speaking exercises to help users master vocabulary, grammar, and conversational skills. Designed for both locals and visitors, it is an inclusive platform that promotes multilingualism and cross-cultural communication.

**Target Audience:** The primary audience includes South African citizens, tourists, expatriates, and students interested in learning local languages for cultural enrichment, professional growth, or personal connection (CivicScience, 2023).

**Additional Features:** (Lingo, 2024)

* Progress-Based Learning: Instead of game-like rewards, users earn points by completing tests. They can track their individual progress, seeing how many words and phrases they got correct. This makes learning motivating by showing clear growth over time.
* Leaderboards & Rankings: Users can see their progress compared to others using the app. It’s innovative because it adds friendly competition, boosting motivation and community spirit.
* Daily Targets & Reminders: Learners set daily goals and get reminders to stay on track. It’s innovative because it helps build a consistent learning habit despite busy schedules.
* Audio-Based Learning: When learning, users can listen to words, phrases, and quotes being pronounced through built-in speakers. This helps them hear and learn the correct pronunciation, improving their listening and speaking accuracy.
* AI Chat Support: Users can talk to an AI chatbot to ask how to say certain words or phrases. The chatbot gives both the written word and a clear pronunciation guide (e.g. Hallo (HAH-loh), helping users learn how to speak naturally.

**Functional Requirements** (AltexSoft, 2023)

**User Registration and Single Sign-On (SSO):** Users can register or log in using Google. Registration collects basic details like name, email, language and password. SSO allows fast, convenient access without repeated credential entry.

**Biometric Authentication:** Optional fingerprint or facial recognition provides secure and quick login, complementing standard login methods.

**User Profile and Settings:** Users can navigate through sections like Home, Language, Words, Phrases, Leaderboard, Visibility Modes, Settings, and Profile. Within Settings, they can manage their profile information, notifications, privacy and security, visibility modes, help and support, view app information (About), and log out — creating a fully personalised and controlled experience.

**Gamification Features:** When users complete their daily goals for words or phrases, a gamification pop-up appears to celebrate their achievement. Completing both daily goals unlocks a special combined gamification pop-up, making progress feel rewarding and encouraging consistent learning.

**Leaderboard and Rankings:** Leaderboards show users’ progress compared to other users using the app. Rankings update in real-time based on test results.

**Daily Targets and Reminders:** Users can set personal goals and receive notifications if they fall behind, helping them stay consistent with learning.

**Speech-Based Learning:** The system must provide audio playback for all words, phrases, and quotes, allowing users to listen to correct pronunciations. This functionality should help users improve their listening and speaking accuracy. **Offline Mode with Sync:** Users can play an offline quiz game to practice lessons without internet access. Their earned experience points and progress automatically sync with their account once they are online, ensuring learning continues even offline.

**Push Notifications:** Notifications remind learners about lessons, goals, and new content to maintain engagement.

**Multi-Language Support:** The app interface supports all seven South African languages, ensuring accessibility for diverse learners

**API Integration:** The system must connect to a REST API hosted to power the AI chatbot, enabling users to ask how to say and pronounce words and phrases.

**Non-Functional Requirements** (Perforce, 2025)

**Performance:** The app should load lessons quickly and support multiple users without lag. Offline functionality must work reliably.

**Security:** User data is encrypted. Biometric login and HTTPS for API communication ensure security.

**Usability:** The interface is intuitive and easy to navigate, suitable for learners with different digital skills.

**Scalability:** The app must handle growth in users, lessons, and gamification data without affecting performance.

**Compatibility:** The app runs on Android 8.0+ and adapts to various screen sizes and orientations.

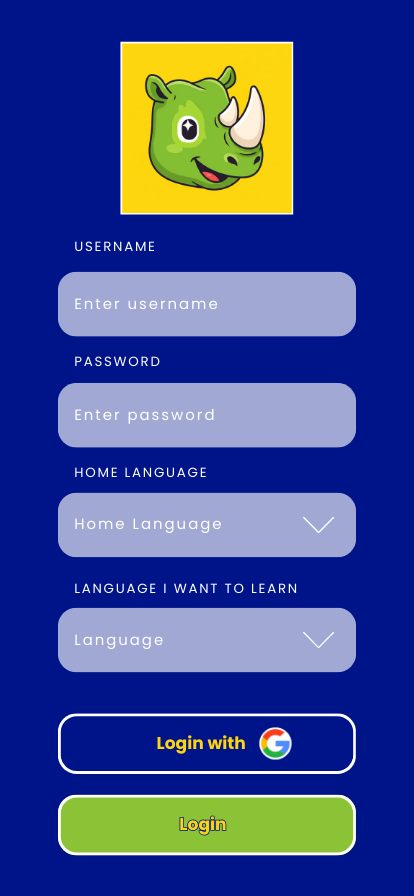
**Technical/System Requirements**

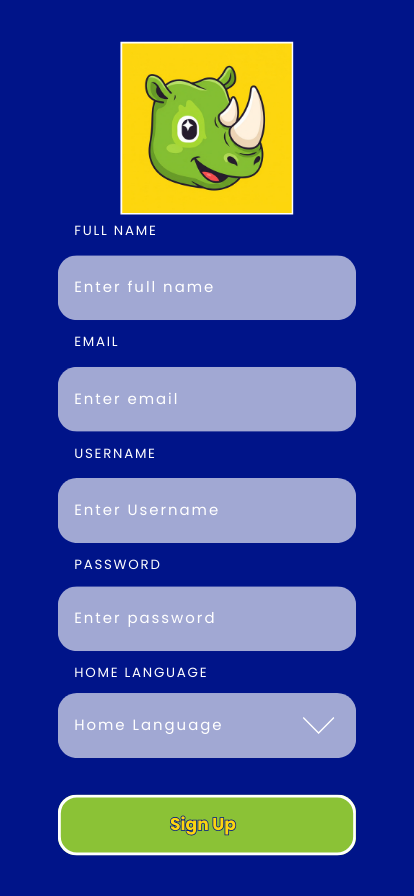
**Development Tools:** Developed in Android Studio using Kotlin or Java. Offline data stored using RoomDB or SQLite.

**Data Management:** Stores user info, lesson progress, points, and leaderboard rankings. Lesson content and responses use JSON for backend interaction.

**Testing and Automation:** Unit tests cover login, lesson completion, and offline sync. GitHub Actions automates testing for continuous integration.

**Hosting and Backend:** The AI chatbot is powered by a REST API hosted on Railway, ensuring scalability and secure handling of user queries related to pronunciation and language assistance.

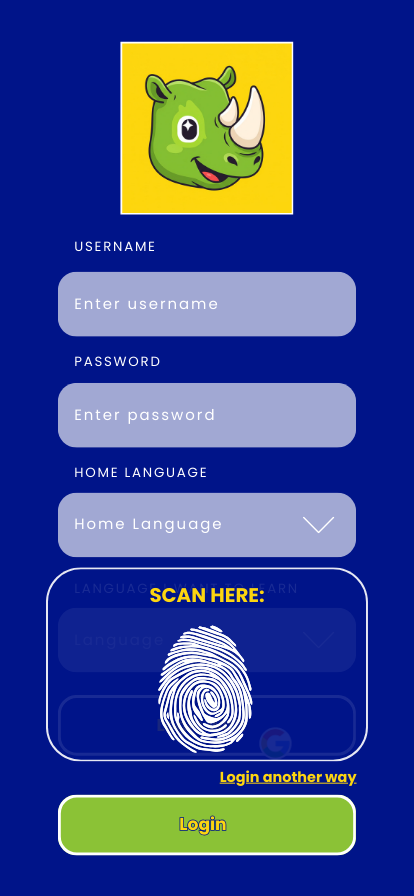
**Mzansi Lingo User Interface:**

****

Login page

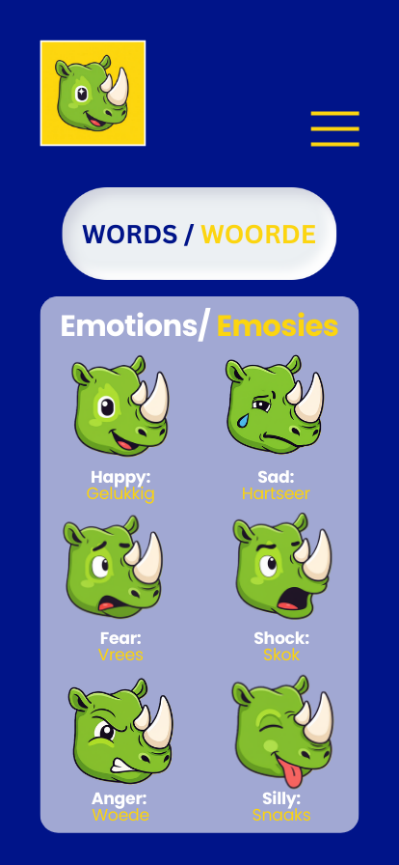
Sign up page

****



Gamification

Biometrics

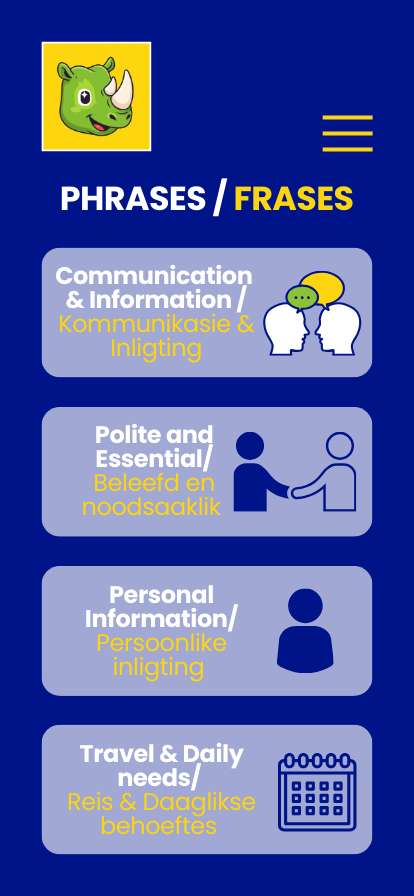
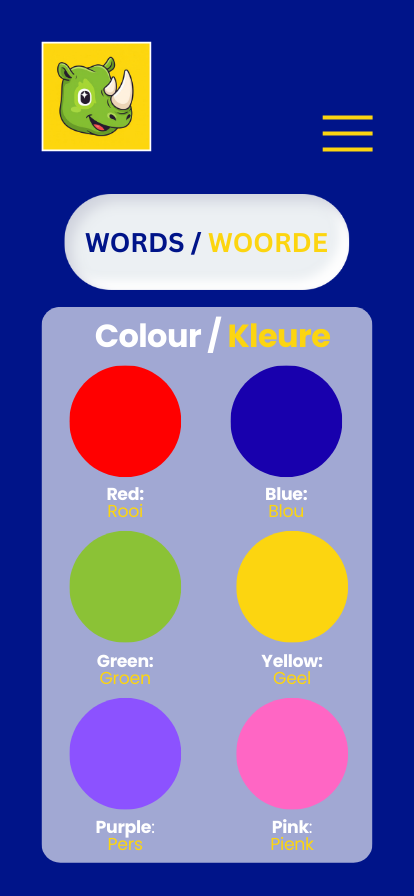
********

Words Page (Emotions)

Words Categories

Navigation Bar

Home Page

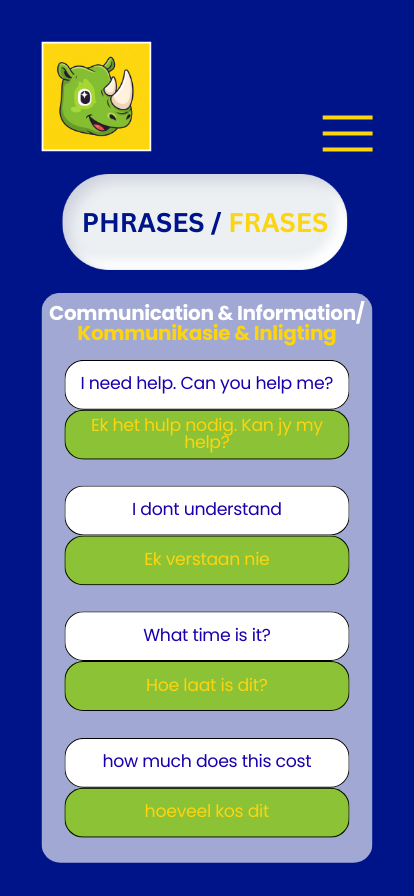
********

Phrases Page (Politeness)

Phrases Categories

Tests for words

Words Page (Colors)

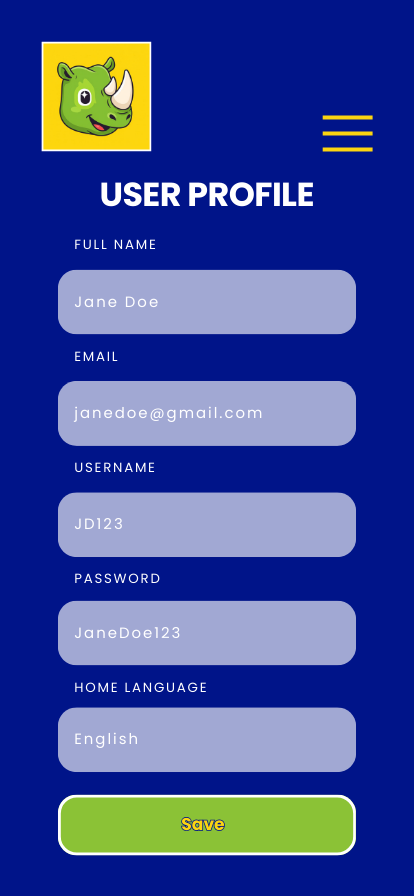
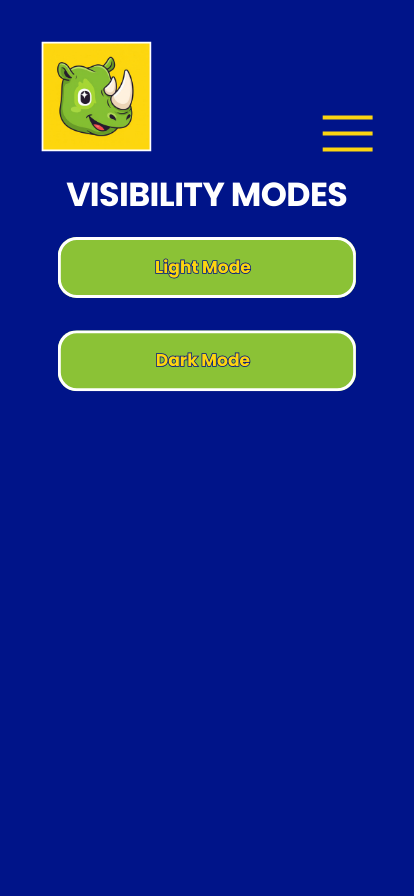
********

Progress Page

Settings Page

Phrases Page (Communication)

Tests for Phrases

****

User Interface Link : <https://www.canva.com/design/DAGwIvHEvPk/2gy2JF65d9vIGbYore9R3Q/edit?utm_content=DAGwIvHEvPk&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton>

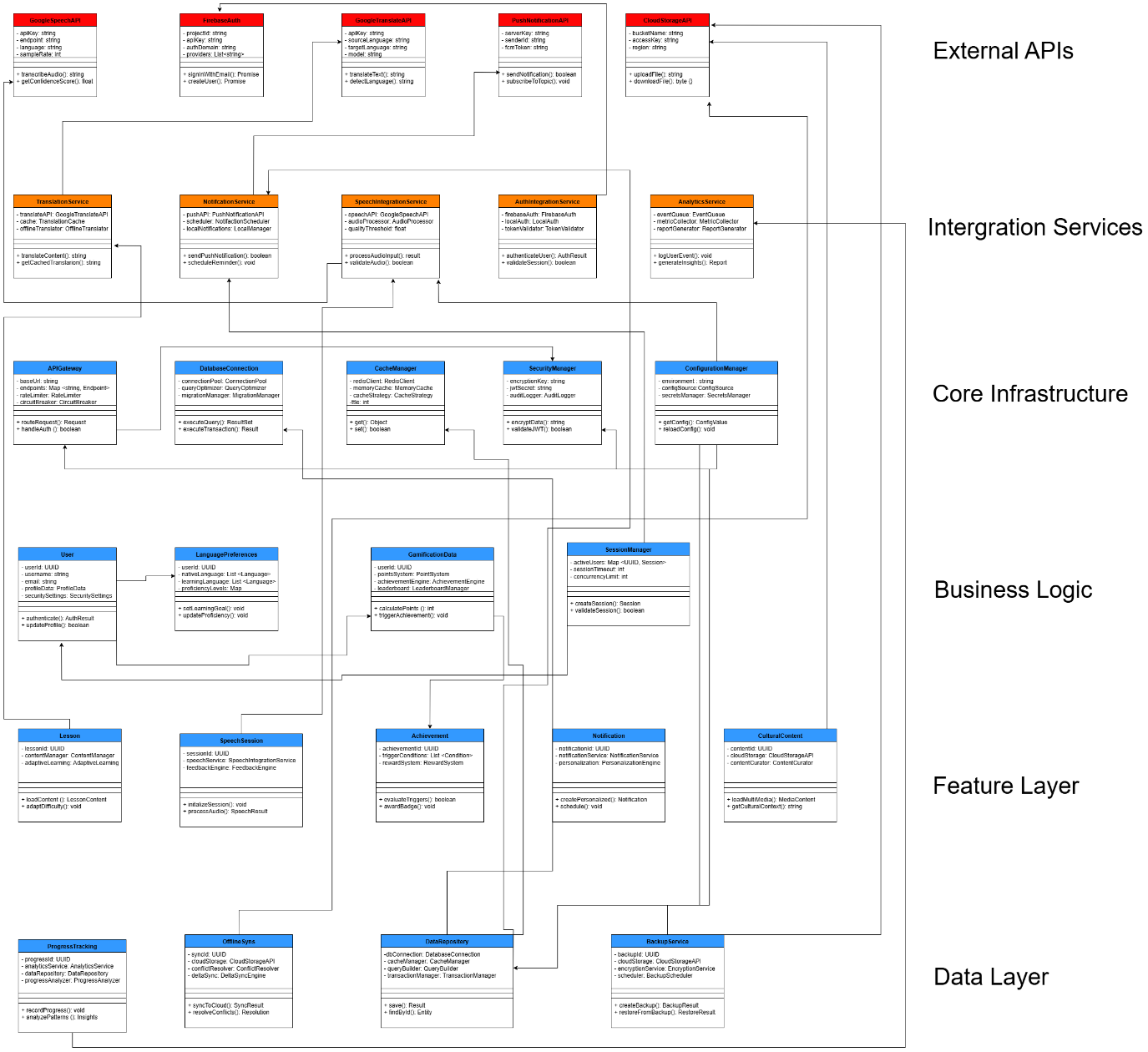
(Canva Pty Ltd 2013)

Visibility Mode

User Profile

**UML Diagram**

**UML diagram link: https://drive.google.com/file/d/1DLMGjPWB\_WUoG7MHNhrL40nRVI\_Z9e1g/view?usp=sharing**

****

**REST API**

The conversational component of Mzansi Lingo is intended to allow users to participate in interactive language practice sessions in their preferred South African language. Instead of chatting with other users, students communicate directly with the system, which delivers text or voice answers, pronunciation checks, and grammatical comments. The technology logs each session, allowing users to review past chats for modification and progress monitoring. Conversation sessions are saved in a special database, which includes elements like session identification, user identity, language of practice, method of communication, and timestamps for tracking progress over time. Each communication in a session is logged, including the sender, message type, message content, and structured feedback on fluency and grammar, resulting in a clear and trackable learning path (AltexS+oft, 2023).

For implementation, the system makes use of cloud-based services and natural language processing technology. Speech recognition is accomplished via technologies like Google Speech-to-Text or similar services, which turn spoken words into written text for analysis. Text-to-speech services are incorporated to offer learners with spoken feedback in their target language, creating a more immersive experience. Speech processing services provide pronunciation testing and grading, allowing users to get constructive feedback on their performance in real time (Krüger, 2025). All text-based chat logs are kept in Firebase Firestore, guaranteeing safe and scalable user data management, and audio recordings are transferred to Firebase Storage, a low-cost and dependable media storage solution. To sustain engagement, notifications of new answers and session changes are sent using Firebase Cloud Messaging, which provides seamless mobile integration and low latency (Arambepola, 2024).  
  
The system is hosted on a lightweight cloud infrastructure. Serverless computing platforms like Firebase Cloud Functions and free-tier services like Heroku are used to handle request processing, session management, and external service interaction. This strategy offers scalability while being cost-effective and dependable, especially throughout the prototype and early adoption stages (CivicScience, 2023). In this approach, the conversational system guarantees that learners may practise speaking, listening, and reading abilities in an interactive environment that is both adaptable and culturally appropriate to South African settings.

**DATA list**

1. User Data

| Field | Data Type | Description |
| --- | --- | --- |
| user\_id | UUID / String | Unique identifier for each user |
| first\_name | String | User’s first name |
| last\_name | String | User’s last name (optional) |
| username | String | Public display name |
| email | String | Email for login and recovery |
| phone\_number | String | Optional; for verification |
| password\_hash | String | Securely stored hashed password |
| profile\_picture | String (URL) | Profile picture |
| date\_of\_birth | Date | Used for age-related access |
| gender | Enum (Male/Female/Other/Prefer not to say) | Optional demographic |
| bio | Text | Short description of user |
| location | GeoCoordinates (Latitude, Longitude) | For community & nearby matches |
| account\_created | Timestamp | Account creation time |
| last\_login | Timestamp | Last login date/time |
| login\_method | Enum (email, google, facebook, biometric) | SSO / login option used |
| consent\_location | Boolean | Permission for location-based matching |
| consent\_data\_usage | Boolean | Permission for storing and analyzing learning/audio data |

2. Language & Learning Preferences

| Field | Data Type | Description |
| --- | --- | --- |
| native\_languages | Array of Strings | User’s native language(s) |
| learning\_languages | Array of Objects | Each: {language, level: Beginner/Intermediate/Advanced/Fluent} |
| interface\_language | String | App interface language |
| daily\_goal | Integer | Number of minutes/lessons set by user |
| reminders\_enabled | Boolean | If user wants daily notifications |
| preferred\_practice\_mode | Enum (text, voice, quiz, conversation) | Chosen practice mode |
| availability | JSON | Time slots user is available for practice |

3. Gamification Data

| Field | Data Type | Description |
| --- | --- | --- |
| points | Integer | Total experience points (XP) |
| badges | Array of Strings | Badges earned (e.g., “7-day streak”) |
| streak\_count | Integer | Consecutive days of activity |
| level | Integer | Current level in app progression |
| leaderboard\_rank | Integer | User’s position on leaderboard |
| achievements | JSON | Completed milestones with timestamp |

4. Lesson & Progress Data

| Field | Data Type | Description |
| --- | --- | --- |
| lesson\_id | UUID | Unique lesson identifier |
| language | String | Language of lesson |
| title | String | Lesson name (e.g., “Greetings in isiZulu”) |
| content | JSON | Lesson content (text, audio, exercises) |
| completed | Boolean | Whether lesson is finished |
| progress\_score | Integer (0–100) | User’s mastery score for the lesson |
| last\_accessed | Timestamp | When lesson was last opened |
| offline\_progress | JSON | Lessons done offline (synced later) |

5. Speech & AI Conversation Data

| Field | Data Type | Description |
| --- | --- | --- |
| session\_id | UUID | Unique conversation session |
| user\_id | UUID | Linked user |
| language | String | Practiced language |
| mode | Enum (voice, text) | Mode of practice |
| conversation\_log | JSON | Chat/voice transcript with timestamps |
| audio\_recordings | Array of URLs | Stored voice clips |
| pronunciation\_score | Float (0–1) | AI feedback on pronunciation |
| fluency\_score | Integer (0–100) | AI-assigned fluency rating |
| feedback\_summary | Text | AI’s critique: strengths + improvement tips |

6. Community & Peer Matching Data

| Field | Data Type | Description |
| --- | --- | --- |
| match\_id | UUID | Unique peer match identifier |
| user\_id\_1 | UUID | First user |
| user\_id\_2 | UUID | Second user |
| shared\_languages | Array of Strings | Common exchange languages |
| distance\_km | Float | Distance between users |
| match\_status | Enum (pending, accepted, blocked) | Status of match |
| friend\_list | Array of User IDs | Saved peer connections |

7. Messaging & Social Interaction Data

| Field | Data Type | Description |
| --- | --- | --- |
| chat\_id | UUID | Chat session ID |
| message\_id | UUID | Message identifier |
| sender\_id | UUID | Message sender |
| receiver\_id | UUID | Message recipient |
| message\_type | Enum (text, voice, image) | Message content type |
| message\_content | Text / URL | Content or file link |
| timestamp | Timestamp | When message sent |
| status | Enum (sent, delivered, read) | Delivery/read status |

8. System, Security & Payments

| Field | Data Type | Description |
| --- | --- | --- |
| device\_info | JSON | Device type, OS version |
| biometric\_enabled | Boolean | Fingerprint/face login enabled |
| subscription\_plan | Enum (free, premium, trial) | Current subscription plan |
| payment\_history | Array of Objects | Past payments {date, amount, method} |
| error\_logs | JSON | Errors/crash reports |
| reported\_users | Array of User IDs | Users reported for misconduct |

****

A screenshot of a computer

AI-generated content may be incorrect.



**Link for Project plan:** [Project plan](https://1drv.ms/x/c/2ba3c5b3edd4ef11/EU-8rllfrxBPolmE_3RH8DgBn2tXsJ0xdvdxI5GZ4XLZFg?e=bcRwao&nav=MTVfezAwMDAwMDAwLTAwMDEtMDAwMC0wMDAwLTAwMDAwMDAwMDAwMH0) **Project plan** (excel, 2025)

**Conclusion**

In summary, the design of Mzansi Lingo reflects a balance between technical innovation and cultural inclusivity. By incorporating features such as pronunciation assessment, conversation logging, and real-time feedback, the system provides a structured yet flexible approach to language learning. The use of cloud-based services ensures scalability and security while maintaining accessibility for a wide audience. Through its conversational functionality, Mzansi Lingo supports learners in developing confidence and fluency, ultimately promoting multilingualism and cross-cultural communication in South Africa (Arambepola, 2024; Krüger, 2025).**Application of AI Tools in the Assessment**

During the course of this assessment, Artificial Intelligence (AI) tools were used as supportive resources to improve efficiency and enhance the overall quality of the work.

**1. Research and Information Support**  
AI tools were especially useful in the early stages of the project when gathering information. Instead of manually navigating through numerous sources, AI provided concise summaries, explanations, and examples that offered a solid foundation for further exploration. This made it easier to understand how related projects had been approached and to identify established best practices. All information obtained from AI was carefully verified against credible sources, ensuring accuracy and maintaining academic integrity.

**2. Visuals and Image Creation**  
Another way AI contributed was in the creation of visuals. Image-generation tools were used to design graphics that matched the project’s goals and aesthetic. For example, when playful illustrations were needed, AI quickly produced creative, child-friendly designs. These images enhanced the presentation of the assessment while saving time and resources that would normally be required for professional design work. The AI-generated outputs were then refined and tailored to suit the specific context of the project, showing how AI can be integrated as a creative partner rather than just a tool.

In summary, AI tools added value to this assessment by streamlining research, providing creative design support, and improving the final presentation. They were used responsibly and ethically, serving as aids to the process without taking away from the originality or integrity of the work. All outputs were reviewed, adapted, and properly acknowledged to ensure transparency. By supporting research and visual design, AI made it possible to produce a more engaging and effective final product.

**References**

* AltexSoft Editorial Team, 2023. Functional and Nonfunctional Requirements: Specification and Types. [online] AltexSoft. Available at: <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/> [Accessed 15 Aug 2025].
* Arambepola, D., 2024. 5+ Must Have Features In Language Learning Apps: A Comprehensive Guide. [online] Ling.app. Available at: <https://ling-app.com/blog/language-learning/> [Accessed 15 Aug 2025].
* CivicScience, 2023. The Word on Language Learning Apps – What’s Behind Their Growing Popularity? [online] CivicScience. Available at: <https://civicscience.com/the-word-on-language-learning-apps-whats-behind-their-growing-popularity/> [Accessed 14 Aug 2025].
* Krüger, G., 2025. Non-Functional Requirements: Tips, Tools, and Examples. [online] Perforce. Available at: <https://www.perforce.com/blog/alm/what-are-non-functional-requirements-examples> [Accessed 15 Aug 2025].
* Microsoft, n.d. Excel. [online] Microsoft. Available at: <https://excel.cloud.microsoft/> [Accessed 14 Aug 2025].
* Canva Pty Ltd. 2013. Canva [software]. Canva Pty Ltd. Available at: <https://www.canva.com/> (Accessed: 25 August 2025).
* AltexSoft Editorial Team, 2023. Functional and Nonfunctional Requirements: Specification and Types. [online] AltexSoft. Available at: <https://www.altexsoft.com/blog/functional-and-non-functional-requirements-specification-and-types/> [Accessed 15 Aug 2025].
* Arambepola, D., 2024. 5+ Must Have Features In Language Learning Apps: A Comprehensive Guide. [online] Ling.app. Available at: <https://ling-app.com/blog/language-learning/> [Accessed 15 Aug 2025].
* CivicScience, 2023. The Word on Language Learning Apps – What’s Behind Their Growing Popularity? [online] CivicScience. Available at: <https://civicscience.com/the-word-on-language-learning-apps-whats-behind-their-growing-popularity/> [Accessed 14 Aug 2025].
* Krüger, G., 2025. Non-Functional Requirements: Tips, Tools, and Examples. [online] Perforce. Available at: <https://www.perforce.com/blog/alm/what-are-non-functional-requirements-examples> [Accessed 15 Aug 2025].