**💡 Main Idea**

**Daily conversation practice improves language skills.**  
The app offers a personalized learning experience focused on real-time practice across reading, writing, listening, and speaking.

**🧩 High-Level Components**

**1. Authentication**

* **Purpose**: User login and access control.
* **APIs**: Google Authentication
* **Subcomponents**:
  + Sign-in / Sign-up screen
  + Google OAuth integration
  + User profile creation

**2. Home Screen**

* **Purpose**: Entry point and navigation hub.
* **Fields/Elements**:
  + Language level status
  + Last activity
  + Recommended practice today
  + Navigation to Test / Chat / Games / Settings

**3. Chat (Practice Hub)**

* **Purpose**: Practice language in all parameters with AI.
* **Features**:
  + Text input + feedback
  + Voice input (with speech-to-text)
  + AI response (text/voice)
  + Revise message options
* **Constraints**:
  + Max 7 seconds of recording
  + Max 300 characters per message
* **APIs**:
  + Gemini for intelligent responses
  + DeepSpeech or similar for speech recognition
* **Subcomponents**:
  + **Message Bubble**
    - Text or voice
    - Revise button (if needed)
    - Feedback area (grammar, vocabulary tips)
  + **Input Bar**
    - Text field
    - Record button
    - Send button
  + **Topic Selector**
    - Predefined or random topics
    - Adjusted to level

**4. Language Level Test**

* **Purpose**: Assess user's language proficiency (CEFR-based).
* **Rules**: ~2 minutes per test.
* **Sections**:
  + Reading a phrase + questions
  + Writing task
  + Listening comprehension (AI reads 2 questions)
  + Speaking tasks (repeat and answer)
* **Subcomponents**:
  + **Instruction Modal**
  + **Task Cards**
    - Input area (text/record)
    - AI evaluation logic per skill
  + **Results Screen**
    - Breakdown: Reading, Writing, Listening, Speaking
    - Estimated CEFR level

**5. Settings**

* **Purpose**: Control app behavior, appearance, and preferences.
* **Subscreens/Components**:
  + **Language Selection**
    - App UI language
    - Target language
  + **Theme Settings**
    - Dark Mode toggle
  + **Time Limit**
    - Usage cap per day (e.g., 30 min/day)
  + **Feedback Preference**
    - Level of feedback in chat (basic correction / contextual suggestions / challenge mode)
  + **Payment Method** *(if premium features)*
    - API: Stripe, PayPal, or others

**6. Games for Kids**

* **Purpose**: Language learning through fun activities.
* **Features**:
  + Mini-games focused on vocabulary, basic sentences
  + Voice and image interactions
* **Subcomponents**:
  + Game cards
  + Scoreboard
  + Progress tracker

**⚙️ System Rules**

* Strings externalized for multi-language support (i18n)
* Recording limit: max 7 seconds
* AI prompt/output limit: capped per interaction
* Dark mode support
* Language-level aligned with CEFR standards

**🧠 Learning Doctrine: CEFR**

All learning and feedback are aligned with CEFR:

* A1 → C2 levels
* Skill categories: Reading, Writing, Speaking, Listening

**🧭 Workflow Overview**

**User Journey**

1. **First Time**: Sign in → Take Level Test → Land on Home → Start Chat or Games
2. **Daily**: Home → Chat/Practice → Feedback/Revise → Improved Score
3. **Review**: Progress tracking → Re-test if needed

**🛠️ Tech Stack Suggestions**

* **Frontend**: React Native (multi-platform), Flutter (option)
* **Backend**: Node.js (API Gateway), Firebase or Express
* **AI Integration**: Gemini API, OpenAI (backup), Mozilla DeepSpeech
* **Database**: Firebase Firestore / MongoDB
* **Payment**: Stripe API
* **Localization**: i18n with crowdin or simple JSON structures

**🔨 Suggested Developer Task Splits**

| **Team Member** | **Responsibility** |
| --- | --- |
| Dev 1 | Authentication + User profile + Home Screen |
| Dev 2 | Chat screen + AI integration + Feedback system |
| Dev 3 | Level Test engine + Results breakdown |
| Dev 4 | Settings + Theme + Localization |
| Dev 5 | Games for kids + Mini-game logic |
| Dev 6 | Backend APIs + DB models + Payment gateway |

App

│

├── AuthProvider (Handles Google OAuth, session management)

│

├── LanguageProvider (Handles i18n and language switching)

│

├── ThemeProvider (Dark Mode / Light Mode support)

│

├── Navigation

│ ├── HomeScreen

│ │ ├── UserProgressCard

│ │ ├── DailySuggestionCard

│ │ ├── QuickActions (Chat, Test, Games)

│

│ ├── ChatScreen

│ │ ├── ChatHeader (Level indicator + topic selector)

│ │ ├── ChatHistory

│ │ │ ├── MessageBubble (User)

│ │ │ │ ├── TextContent

│ │ │ │ ├── VoicePlayer (if recorded)

│ │ │ │ ├── FeedbackPanel

│ │ │ ├── MessageBubble (AI)

│ │ │ ├── TextContent / Voice

│ │ ├── ChatInputBar

│ │ │ ├── TextInput

│ │ │ ├── RecordButton

│ │ │ ├── SendButton

│

│ ├── TestScreen

│ │ ├── InstructionModal

│ │ ├── QuestionSection

│ │ │ ├── ReadingCard

│ │ │ ├── WritingTaskCard

│ │ │ ├── ListeningCard

│ │ │ ├── SpeakingTaskCard

│ │ ├── Timer

│ │ ├── SubmitButton

│ │ ├── TestResultScreen

│ │ ├── CEFRLevelDisplay

│ │ ├── SkillScoreCard (Reading, Writing, Listening, Speaking)

│

│ ├── GamesScreen

│ │ ├── GameSelector (List of games)

│ │ ├── GameComponent (varies by game)

│ │ ├── ScoreBoard

│ │ ├── ProgressTracker

│

│ ├── SettingsScreen

│ ├── LanguageSettings

│ ├── ThemeSettings (Dark Mode Toggle)

│ ├── DailyUsageLimit

│ ├── FeedbackLevelSelector

│ ├── PaymentSettings

│ ├── VersionInfo

│

└── Shared Components

├── Modal

├── Loader

├── Snackbar

├── VoiceRecorder

├── AudioPlayer

├── Button

├── Card

├── ToggleSwitch

├── ProgressBar

Backend-

**🔌 Key APIs**

**🧑‍💻 Auth**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| /auth/google | POST | Google OAuth login |
| /auth/logout | POST | Logout and invalidate token |

**💬 Chat**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| /chat/sendMessage | POST | Send text or voice message |
| /chat/getMessages | GET | Get past chat history |
| /chat/feedback/:id | GET | Get AI feedback on a message |
| /chat/revise/:id | POST | Submit a revised version |

**🧪 Language Test**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| /test/start | POST | Begin adaptive 2-minute test |
| /test/submit | POST | Submit answers & audio |
| /test/results/:userId | GET | Get latest test results |

**🎮 Games**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| /games/list | GET | List all available games |
| /games/submit/:gameId | POST | Submit result for a game |
| /games/progress/:userId | GET | Get user's game progress |

**⚙️ Settings**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| /settings/:userId | GET | Get current settings |
| /settings/update/:userId | PUT | Update settings |

**💰 Payments**

| **Endpoint** | **Method** | **Description** |
| --- | --- | --- |
| /payment/subscribe | POST | Subscribe to a plan |
| /payment/status/:userId | GET | Get user subscription status |

**🧬 Database Models**

**User.**js

{

id, email, name, level, settingsRef, testResults[], chatHistory[], gameProgress[],

isPremium, createdAt

}

**ChatMessage.**js

{

id, userId, messageType: 'text' | 'voice', content, aiResponse, feedback, revisedMessage, createdAt

}

**TestResult.**js

{

id, userId, scores: { reading, writing, listening, speaking }, CEFRLevel, createdAt

}

**Settings.**js

{

id, userId, darkMode, language, reviseLevel, dailyLimit, audioLengthLimit, promptLengthLimit

}

**⚙️ Services**

* **aiService.js**: Talk to Gemini, generate responses, test questions, chat topics.
* **speechService.js**: Transcribe user voice and generate AI audio responses.
* **levelService.js**: Based on user inputs, determines CEFR level.
* **feedbackService.js**: Grammar/fluency evaluation using AI.
* **paymentService.js**: Interact with Stripe/PayPal for subscriptions.

**📢 Server-to-Client Communication**

Use **REST** for:

* Auth
* Test submissions
* User settings

Use **WebSocket** for:

* Real-time chat (optional)
* Live feedback

**🌐 Multi-language & Prompt Limitation**

* Use i18nManager.js to serve strings in multiple languages.
* Use promptLimiter.js to check for prompt/output character limits (e.g. max 300 chars, 7-sec audio).

**🔁 Development Teams**

| **Developer** | **Responsibility** |
| --- | --- |
| Dev 1 | Auth & Middleware |
| Dev 2 | Chat backend + Gemini API |
| Dev 3 | Testing logic (CEFR mapping, scoring) |
| Dev 4 | User Settings + Multi-language |
| Dev 5 | Games backend + progress |
| Dev 6 | Audio transcription + feedback services |
| Dev 7 | Payment & Subscription |