

# CipherAura: 1-Year Technical Architecture & Tech Stack Plan

**Vision & Overview:** CipherAura is a **gamified, AI-driven language-learning platform** built on modern principles. The one-year roadmap (Nov 2025–Diwali 2026) lays out phased goals from **foundations to beta launch**. Core themes include *microservices*, *API-first design*, and *privacy-by-design*. Our timeline culminates in a closed-beta launch by Diwali 2026. To ensure clarity, we align each phase with specific tech goals and compliance checkpoints.

## Phase 0: Foundation & Discovery (Diwali '25–Dec '25) – “The Blueprint Phase”

*Icon: Figma (UI/UX prototyping and design)*

In Phase 0, the team finalizes planning and design before writing code. Key activities include *competitive analysis*, *feature benchmarking*, and defining core principles (e.g. microservices, API-first, privacy). We also finalize the “Week-to-Conversational” curriculum and create high-fidelity UI/UX prototypes (using tools like Figma). This blueprint ensures the platform is **future-proof**, **privacy-centric**, and fully scoped before development begins.

- **Competitive Analysis & Benchmarking:** Study other edtech/A.I. language tools to set features and KPIs.
- **Core Technical Principles:** Document standards (microservices, API-first, privacy-by-design).
- **Curriculum Finalization:** Lock down the 1-week learning content arc.
- **UX Prototyping:** Design mockups and user flows (Figma/Adobe XD).

## Phase 1: Landing Page & Waitlist (Jan '26–Feb '26) – “The Teaser”

*Icon: React (Next.js frontend)*

Phase 1 delivers a **hype-building landing page** and email waitlist capture. We use Next.js (React) with Tailwind CSS for a **fast, SEO-friendly front end**. Hosting is on Vercel or Netlify for simple, scalable deployment. The backend is a lightweight Node.js/Firebase API to handle form submissions, and we store data in MongoDB Atlas or PostgreSQL. The goal is to “**sell the vision**” and start building a community of early adopters.

- **Frontend:** Next.js + Tailwind CSS for a responsive, modern web UI.
- **Hosting:** Deployed on Vercel or Netlify (static site/CDN).
- **Backend API:** Simple Node.js/Firebase function to receive sign-ups.
- **Database:** MongoDB Atlas (for speed) with fallback to PostgreSQL (for structure).
- **Milestone:** Launch public landing page → hook up sign-up form → deploy.

## Phase 2: Core Platform & Gamification (Mar '26–Jun '26) – “The Engine”

*Icon: Node.js (backend services)*

Phase 2 builds the **core application and gamification system**, sans complex AI. Users can register, play lessons, and earn rewards. The stack is based on **full-stack JavaScript**: Next.js/React frontend and a Node.js + Express backend. We use PostgreSQL for structured data and Redis for fast session caching or leaderboards. Authentication is via NextAuth.js (with Firebase as an option). Gamification (points, badges, leaderboards, festival-themed challenges) runs in a **separate Node.js microservice** for flexibility. The UI supports text, audio/speech, and quizzes, integrating Web Audio and speech APIs.

- **User Accounts:** Sign-up/log-in via NextAuth.js (JWT/session).
- **Lesson Player:** Interactive modules (15-min/day chunks) with text, recorded audio, speech practice, and quiz widgets.
- **Gamification Service:** A decoupled Node.js microservice for events (points, leaderboards, seasonal quests).
- **Infrastructure:** Host frontend & backend (e.g. on Railway/Render or Heroku for easy scaling).

## Phase 3: AI Integration & Personalization (Jul '26–Sept '26) – “The Brain”

*Icon: PyTorch (AI & ML models)*

Phase 3 integrates AI for **speech assessment, adaptive learning, and recommendations**. We leverage Indic-language models from AI4Bharat (IndicBERT for understanding, IndicTrans2 for translation, IndicWhisper for speech-to-text). These models are hosted via managed services (Hugging Face Inference or Google Vertex AI). We plan to **fine-tune** these models on our learner data. In parallel, a Python-based log-scanning service (e.g. FastAPI) analyzes usage to auto-suggest micro-content and improve the curriculum. This phase ensures each user gets personalized feedback (e.g. pronunciation scores) and content pacing via real-time AI.

- **Indic LLMs & Models:** Start with pre-trained AI4Bharat/HuggingFace models (IndicBERT, etc.).
- **Hosting:** Serve models on Hugging Face Endpoints or Vertex AI (scalable GPUs).
- **Fine-Tuning:** Train on crowdsourced speech/corpus data to improve accuracy in Hindi and other Indian languages.
- **Recommendation Engine:** Use log data + clustering (e.g. scikit-learn) to create spaced-repetition schedules and content suggestions.

## Phase 4: Beta Preparation & Launch (Oct '26) – “The Countdown”

*Icon: Android (React Native mobile)*

Phase 4 wraps up the product for beta release. We build a **cross-platform mobile app** using React Native (Expo). Rigorous testing is performed with ~100–200 users, iterating on feedback. We integrate error monitoring (Sentry) and privacy-friendly analytics (Plausible). Load testing (k6) and performance tuning ensure stability. Crucially, we complete a **DPDP Act compliance review**: privacy features (consent flows,

data export, encryption in transit/at-rest) and appoint a Data Protection Officer for audits. Upon passing security and usability checks, we launch the closed beta in Diwali 2026.

- **Mobile App:** React Native/Expo for iOS/Android deployment.
- **Testing & QA:** Closed beta to refine UX; load-test backend (k6) for expected traffic.
- **Monitoring:** Use Sentry.io and Plausible analytics to track errors and usage in a privacy-first way.
- **Security & Compliance:** Implement DPDP-ready flows (user consent, erasure rights), encryption, breach notification. Conduct a privacy & security audit (appoint DPO).

## Technology Stack Summary

*Icon: Vercel (Cloud hosting & infra)*

Our stack emphasizes modern, scalable tools. Below is a high-level summary (styled as a color-coded table in final PDF):

- **Web Frontend:** *Next.js* (React framework) – fast, SEO-friendly single-page app. Styling with *Tailwind* CSS for rapid UI development.
- **Mobile App:** *React Native* – cross-platform mobile codebase.
- **Backend API:** *Node.js* + *Express.js* – event-driven, non-blocking server. (Future GraphQL microservices planned for content modules.)
- **Databases:** *PostgreSQL* – reliable relational storage for user data and content; *MongoDB* – NoSQL store for flexible data (landing signups, logs); *Redis* – in-memory cache for sessions, leaderboards.
- **AI/ML:** *PyTorch* with Hugging Face Transformers – using AI4Bharat Indic models, fine-tuned for speech/text tasks. Models served via Hugging Face or Vertex AI (GPU-enabled).
- **Cloud & DevOps:** *Containerization* with Docker/Kubernetes for AI inference; *Vercel* (or AWS/GCP) for frontend hosting; early deployment on free-tier hosts (Railway, Render). CI/CD via GitHub Actions for automated builds/deployments.
- **Monitoring & Analytics:** *Prometheus/Grafana* (for services monitoring) and *Sentry.io* (error tracking); *Plausible.io* for lightweight analytics. All tools chosen to respect user privacy (GDPR/DPDP-ready).

## Security & Compliance (DPDP-Ready)

Data privacy is built in from the ground up. We implement **privacy-by-design** features: user consent flows, data portability, and strict data retention policies. All data is encrypted in transit and at rest, with regular breach response drills. We will appoint a Data Protection Officer and conduct annual Data Impact Assessments as required under India's DPDP Act. This ensures CipherAura is not just engaging but **secure and compliant** for learners of all ages.

## Conclusion & Advice

CipherAura blends *playful, culturally-rooted education* with *cutting-edge AI*. Starting with Hindi content, we iterate quickly using community speech data and feedback. By following this phased plan, we launch a robust, gamified language platform by Diwali 2026. Remember: **start simple** (focus on core speech-checker and gamified lessons) and scale up. This architecture and stack ensure a “festival-ready” product that delights learners while keeping privacy central.

**Sources:** Project plan and tech stack details are drawn from the CipherAura Master Project Plan and Roadmap. These informed every recommendation and timeline above.

---