

Since July 2025, I have been working as an **MX Manager at Soundmind**. On a single child safety product, I built all four layers myself, **from the Android native code on the device to the collection server, the parent app, and zero-downtime deployment**.

The problem I wrestled with longest was not losing location data on devices where a management policy keeps mobile data off and logs cannot be read. After funneling data opening and release through a single gateway and correcting for the power-saving timer delays I measured in the field, the **transmission success rate reached 100%**. On the server, I changed the pipeline to drain and store incoming locations atomically and quarantine failures separately, so a single broken record no longer halts the entire load.

In the parent app, I built the logic that interprets noisy coordinates into places stayed and travel routes, eliminating **the problem of indoor stays splitting apart and baseless movement guesses appearing on screen**.

At **WIGTN**, a five-person AI development and research crew, we build AI systems that run in real environments. On **WIGVO**, which performs two-way real-time interpretation over the ordinary phone network, I handled the relay server and deployment infrastructure, recording a median latency of **555 milliseconds** on real calls and **zero echo loops across 147 calls**. The paper was accepted to **ACL 2026 System Demonstrations**.

With the same team, we released **two open-source plugins** covering Claude Code's parallel agent orchestration and Codex's Evidence Contract, and took second place at the **Snowflake AI & Data Hackathon Korea 2026**.

At the company, I have built one product end to end; on the team, I have explored how to put AI into the development process itself. Connecting the two, I want to work on **redesigning how development and operations are done, not just using AI as a tool**.

EXPERIENCE

Soundmind · MX Manager

2025.07 ~ Present

Responsible for the apps, servers, web, and deployment of child safety and education services.

Odiya Child Device App — Solved a problem where transmissions dropped on managed devices that keep mobile data off by default. Funneled data-enable requests through a single gateway and added a minimum 6-second hold rule, securing a 100% transmission success rate.

Odiya Parent App — Designed the location interpretation UI around the principle "better to show nothing than to be wrong". Three-stage subway inference filter, stay clustering, and screen gating by child app version. 89 unit tests.

Odiya Location Collection Server — Stores up to 5,000 records every 60 seconds in batches via a Redis queue, recovering in order on failure. Removed a structure where a single corrupted record halted all storage.

KOCCA Korean Speaking Test Platform — Owned every area: test-taking screens, around 30 grading admin pages, the STT/TTS pipeline, and building the AI training dataset. Destructive schema changes are automatically blocked at deploy time.

Mohani Child Smartphone Usage Management — Designed a structure where devices catch up on settings by themselves even when push messages are lost. Added redundancy so blocking persists even if accessibility permission is turned off.

Unified Login Server — Separated token policies by user type. If a stolen token is reused, every token for that account is invalidated immediately.

Deployment and Operations — Built zero-downtime deployment for three products. Set up Jenkins automated deployment and server/log monitoring.

SELECTED PROJECTS

Odiya Child Device App — An app that sends location without missing a beat, on managed devices where mobile data is normally switched off.

Odiya Parent App — A map app that presents uncertain location data without overstating it. It shows places stayed, movement paths, and real-time tracking.

KSTT Korean Speaking Test Platform — A full-stack Next.js service where I owned everything from the test-taking screens to the grading admin, speech recognition and synthesis, and building the training dataset.

WIGVO Real-Time Phone Call Interpretation — Interprets regular phone calls in both directions in real time. The receiving side does not need to install an app. I participated as a co-author of a paper accepted to ACL 2026.

AWARDS · PUBLICATIONS

- 2022 **JLPT N1 (Japanese-Language Proficiency, Level 1)** · Business-level still developing; reads kanji largely by context.
- 2026 **Snowflake 2026 Hackathon · Runner-up** · wightn-for-snowflake · Team WIGHTN CREW
- 2026 **ACL 2026 Demo Track · Accepted** · wigvo-v2 · Team WIGHTN CREW

EDUCATION

- Gangseo Polytechnic College** Smart Finance Program (left early after job offer) 2025.03 ~ 2025.06
- University of Ulsan** Chinese Lit / Japanese Lit (B.A.) 2016 ~ 2024

SKILLS

- React Native · Next.js · TypeScript · Spring Boot · MariaDB · Redis · Docker · React · Android (Java/Kotlin) · Expo · Tailwind CSS · TanStack Query · Java · NestJS · Node.js · Prisma · MySQL · PostgreSQL · Nginx · Jenkins · GitHub Actions · GCP Cloud Run