

AI Product Engineer with 6+ years of experience building the interfaces of AI products in React and TypeScript: state design for streaming LLM chat, browser-agent automation with verifiable results, and rendering performance and server-driven UI at scale. On my current team I am the frontend lead for AI chat and the de facto part lead, setting technical direction, reviewing code, and mentoring.

From HCI and Visual Analytics research to explainable-AutoML interfaces and contributions to LLM observability tooling, I keep working on one problem: making what AI systems produce understandable and verifiable, so people can trust and act on it.

Experience

Levvels Frontend Engineer

Jul 2025 – Present

Acted as part lead of a three-person frontend team regardless of title, setting technical direction, reviewing code, and mentoring. Frontend lead for an AI character chat product, designing its conversation UI and streaming state ahead of launch; also built Vuddy commerce and browser-agent automation for CS/QA. Planned and led a company-wide internal talk, AI Agent Study 101, to help every team put AI to real use.

React · Next.js · TypeScript · TanStack Query · Zustand · SSE · Playwright · Sentry · Mixpanel · ChannelTalk · PageAgent · Generative UI

AI Character Chat Service

May 2026 – Present

Led the frontend of an AI character chat product preparing to launch in Korean, Japanese, and English. Owned chat, story detail, and support inquiries, and built the Admin for authoring and reviewing character content.

- Designed the conversation runtime as an eight-state machine (sending, persisted, streaming, completed, send failed, stream failed, blocked, and idle); TanStack Query holds server state, Zustand holds runtime state, and token deltas never enter the Query cache
- Kept each turn provisional until the DONE event; treated five failure classes (prohibited content, network, engine timeout, internal error, safety filter) and streams closed without DONE as failures, rolling back provisional messages while preserving the user's draft
- Blocked overlapping sends and regenerations with single-flight, cancelled stale refetches with a request sequence, and kept server snapshots from overwriting an active stream; verified the runtime hook and reducer with 55 tests
- Derived 61 QA scenarios from 25 Figma policy nodes and automated 10 of them as Playwright E2E tests named by scenario ID
- Wrote 10 frontend specs and 23 ADRs documenting design decisions, and found and catalogued API contract gaps while generating the SDK from Swagger, including 11 operations with untyped responses
- Built authoring and review tools for stories, lorebooks, and search keywords

PageAgent-based Internal Operations Automation

Mar 2026 – Present

CS/QA checks needed developers, planners, and operators to confirm asynchronously: ten minutes at best, half a day often, and after someone's vacation at worst. A browser agent now finishes them in about a minute. Used by three operators, two QA engineers, and three planners.

- Ran Vuddy's 80-case sanity checklist and 19 route-level machine specs through the agent; deterministic checks on data-qa selectors decide each verdict, and a person confirms the final PASS/FAIL
- Automated how operators investigate Google Play in-app payments that were charged but left as failed because the app never reported completion to the server
- Gathered account, payment, and mapping data from several internal tools into one comparison view, separating LLM summaries from execution evidence so operators can re-check every source
- Forked Alibaba PageAgent into a Chrome extension deployed through Chrome Enterprise policy; the agent pauses and asks when a person must decide, such as which login provider or account to use

Vuddy Commerce

Jul 2025 – Present

Built and maintained Vuddy's customer web, Creator Studio, and internal Admin. Worked across the commerce flow from class discovery and checkout to orders, cancellations, returns, and exchanges.

- Built a generative-UI AI dashboard for the Admin: natural-language requests become query plans validated by Zod schemas, and the values the LLM may choose are limited to an allowlist enum derived from API types instead of a blocklist
- Built class discovery, video, and checkout flows along with Studio order management and Admin tools for products, students, and passes
- Updated the cart, identity verification, secondary-creator onboarding, and cancellation, return, and exchange flows as policies and APIs changed
- Fixed HLS home-carousel and mobile WebView issues involving safe areas, scrolling, and back navigation

RIDI Corp. Frontend Engineer

May 2022 – Jul 2025

Member of an eight-person frontend team. Took part in the team's server-driven UI and incremental migration work, which decoupled operational UI changes and A/B tests from release cycles. Interviewed engineering candidates.

Next.js · React · TypeScript · Emotion · Jest · PHP · Twig · Sentry · Virtualization

RIDI Web Platform [↗](#)

May 2022 – Jul 2025

Helped move business-critical pages such as the genre home and title home to React incrementally with server-driven UI and islands. UI changes and A/B tests that took about a week because they needed a release now ship the same day through configuration.

- Separated UI composition from business data with GraphQL Grid/Cell interfaces and reconstructed the tree on the client
- Extended schemas while preserving compatibility with existing clients; reused the structure across landing, search, and detail pages
- Introduced React islands into PHP/Twig pages to migrate incrementally without a full rewrite
- Cached UI responses with Redis and reduced the coupling between operational UI experiments and frontend releases
- Extended the approach to marketing landing, search, and title-detail pages, and used a middleware bottleneck found during partial migration to make the case for full migration

Large-scale List Rendering Optimization [↗](#)

May 2022 – Jul 2025

A screen designed for 250 titles had to handle 6,000+, and its initial render took over 10 seconds. Reworked it with virtualization, dynamic heights, and scroll restoration so it renders without perceptible delay.

- Separated network-transfer costs from CPU and DOM-rendering costs and virtualized the list to render only the required range; confirmed gzip could cut large JSON payloads to about one-eighth
- Handled dynamic heights and scroll restoration with window-based scrolling
- Used SVG-background skeletons to separate loading presentation from content rendering

User Environment Stabilization

May 2022 – Jul 2025

Reduced instability caused by differences in browsers, devices, and extensions, and improved operational response.

- Tracked real-user errors with Sentry
- Handled translation-plugin conflicts
- Improved E-ink rendering and UX issues

TmaxEnterprise Researcher

Feb 2020 – May 2022

On a six-person team, designed product flows, visualizations, and interactions that let non-experts use AI and analytics, and built UIs that show predictions together with their reasons. Started reviewing the team's code here.

React · TypeScript · Material-UI · Sass · Python · jQuery

Legacy System React Migration

Jan 2021 – May 2022

Moved a tightly coupled internal UI library toward React to create a more consistent and extensible foundation for platform interfaces.

- Modularized legacy UI code and improved interface structure
- Developed reusable components to replace bespoke legacy widgets

AutoML Platform [↗](#)

Feb 2020 – May 2022

Designed and implemented codeless studio flows and explainable AI visualizations so users without model-development experience could explore and operate AI capabilities.

- Implemented a codeless studio for non-expert model builders
- Visualized global feature importance, local prediction explanations, and What-if flows, and shipped a prediction-explanation UI showing the prediction, rule path, and feature contributions together

- Researched interfaces between AutoML engines and model developers
- Clarified requirements for an early-stage platform

Open source

Aethrion Author

May 2026 – Present

Building a social simulation runtime in Elixir for persistent AI characters; released v0.1.0 alpha.

- Deterministic, testable rules own state, memory, and relationship changes; the LLM is only an expression layer
- State advances even when an LLM call fails, and an LLM response changes the world only by re-entering as a new event that passes the rules

OpenMMO Contributor

Jul 2026 – Present

Contributing to an open-source MMORPG where AI agents and humans play through the same protocol (Rust · Svelte · WebAssembly); all 10 PRs merged.

- Rust server: account-level bans, terrain-cache eviction with capacity limits, and refusing trades with sleeping merchants
- Client: a HUD minimap scoped with the maintainer, test coverage for fishing and click intents, and a CONTRIBUTING guide for new contributors

vite-plus Contributor

Aug 2026 – Present

Contributing to VoidZero's Vite and Vitest toolchain (Rust · TypeScript).

- Fixed vp env use not applying after the global -C flag and locked the four shell integration templates with snapshot tests (merged)
- Fixed environment variables that invalidated the cache on every GitHub Actions run (vite-task, in review); reviewed a migration PR and co-authored its PTY regression test

Langfuse Contributor

Sep 2026 – Present

Contributing to the React and TypeScript frontend of an LLM observability and evaluation platform.

- Fixed duplicated, undeletable characters during Korean IME composition in the Tracing search bar and added five regression tests (in review)

Education

Ulsan National Institute of Science and Technology (UNIST)

Government-funded research university and one of Korea's four science and technology institutes, alongside KAIST, with a mandate for advanced research, science and engineering education, and technology transfer and startups.

M.S. research in HCI and Visual Analytics.

Master of Science in Computer Science

Mar 2018 – Feb 2020

Bachelor of Science in Technology Management (Multi-disciplinary major in Computer Science)

Mar 2013 – Feb 2018

Publications

1. An Empirical Analysis on Transparent Algorithmic Exploration in Recommender Systems

Kihwan Kim · A Computing Research Repository (CoRR), 2108.00151, 2021

2. ST-GRAT: A Novel Spatio-temporal Graph Attention Networks for Accurately Forecasting Dynamically Changing Road Speed

Cheonbok Park, Chunggi Lee, Hyojin Bahng, Yunwon Tae, Kihwan Kim, Seungmin Jin, Sungahn Ko, Jaegul Choo · ACM International Conference on Information and Knowledge Management (CIKM), 2020

3. Modeling Exploration/Exploitation Decisions through Mobile Sensing for Understanding Mechanisms of Addiction

Kihwan Kim, Sanghoon Kim, Chunggi Lee, Sungahn Ko · ACM International Conference on Mobile Systems, Applications, and Services (MobiSys), 2019

4. An Empirical Study on the Relationship Between the Number of Coordinated Views and Visual Analysis

Juyoung Oh, Chunggi Lee, Hwiyeon Kim, Kihwan Kim, Osang Kwon, Eric D. Ragan, Bum Chul Kwon, Sungahn Ko · A Computing Research Repository (CoRR), 2204.09524, 2018

5. A Survey of Visualization Techniques for Interpreting Deep Learning

Jaesung Lee, Kihwan Kim, Chunggi Lee, Sungahn Ko · Noise & Vibration, Vol.27 No.6, 2017:11

Full experience, case studies, and all open-source contributions are available at kihwan.kim.