

PHAN XUAN CHIEN

Senior Frontend Developer / Software Engineer

Phone: 0374463636 | Email: phanxuanchien183@gmail.com | Portfolio: <https://www.chien-portfolio.blog> |
LinkedIn: <https://www.linkedin.com/in/chien-phan/>
Ho Chi Minh City, Vietnam

SUMMARY

Senior Frontend Developer / Software Engineer with 5+ years of experience specializing in React.js, Next.js, TypeScript, and scalable enterprise web applications. Experienced as a Frontend Owner / Sole Frontend Developer building projects from scratch — including high-fidelity corporate marketing sites (Figma-to-code), role-based partner portals, e-commerce platforms, AI chatbot widgets, POS/kiosk systems, and enterprise workflows. Also building a full-stack Attendance & Payroll HRM product end-to-end (Admin / HR / Employee portals) with Node.js, Express, Prisma, and PostgreSQL — covering multi-portal JWT auth, GPS/Wi-Fi attendance, approval workflows, real-time notifications (WebSocket), and Vietnam-aware payroll (PIT / Social Insurance / statutory allowances). Strong in frontend architecture, pixel-accurate responsive UI, API/auth integration, performance optimization, CI/CD (GitHub Actions / AWS S3 / CloudFront), and unit testing with Vitest, React Testing Library, jsdom, and MSW. Comfortable collaborating with Korean PM/design/backend teams in English and translating Figma/design-system/API notes into production-ready features.

CORE COMPETENCIES

- Frontend ownership: project setup, architecture, routing, reusable components, design-system implementation, code standards, and handover
- React / TypeScript development: React 17+, React 19, Next.js, Vite, typed components, hooks, maintainable code, and performance tuning
- Enterprise & marketing UI: Figma-to-code, pixel-accurate PC/mobile layouts, scroll/viewport animations, reusable table/form patterns, route guards, Tailwind CSS, and Ant Design
- State, API & authentication: Redux Toolkit, Zustand, TanStack Query, REST APIs, Axios clients/interceptors, multi-portal JWT/cookie sessions, refresh-token flow, SSE streaming, and WebSocket real-time notifications
- Testing & quality: Vitest, React Testing Library, jsdom, MSW, unit tests for helpers, stores, hooks, APIs, components, and key screens
- Delivery & collaboration: GitHub Actions CI/CD, AWS EC2/S3/CloudFront (OIDC deploy), Nginx, Docker / Docker Compose, Vercel, and English collaboration with Korean PM/design/backend teams
- Backend & full-stack: Node.js, Express 5, Prisma ORM (v7), PostgreSQL, Swagger/OpenAPI, Zod validation, ExcelJS export, bcrypt/JWT auth, soft-delete audits, payroll calculation modules, and production-oriented API design

TECHNICAL SKILLS

Languages: JavaScript (ESNext), TypeScript

Frontend: React.js, React 19, Next.js, Vite, React Router 7, Web Components, Shadow DOM, i18next (EN/VI), FullCalendar, ApexCharts, scroll/intersection animations

State & Forms: Redux Toolkit, Zustand, TanStack Query, React Hook Form

UI & Styling: Tailwind CSS 4, Ant Design, Bootstrap, Pretendard/custom fonts, responsive PC/mobile UI, Figma/design systems, pixel-accurate marketing layouts

API, Auth & Integration: RESTful APIs, Axios interceptors, JWT authentication, cookie sessions/refresh tokens (multi-portal), SSE APIs, WebSocket notifications, QR identity flows, PortOne identity/payment integration

Testing: Vitest, React Testing Library, jsdom, MSW

Backend, DevOps & Tools: Node.js, Express, Prisma 7, PostgreSQL, Swagger, Zod, ExcelJS, WebSocket (ws), Docker Compose, GitHub Actions, AWS EC2/S3/CloudFront, Nginx, Vercel, oxlint, Figma

PROFESSIONAL EXPERIENCE

BLUE DRAGON COMMUNICATIONS — Frontend Developer | Ho Chi Minh City, Vietnam

Jan 2025 – Present

1. Prototie Corporate Website (prototie.net) (Jul 2026 – Present)

Team: Sole Frontend Developer | **Role:** Frontend Developer / Frontend Owner

Overview: High-fidelity company marketing SPA for Prototie — corporate landing, project showcase, About, IT Outsourcing, IT Solution, and AI Matching service pages — built from scratch to match Korean Figma specs with premium dark-theme UI and full PC/mobile responsiveness.

Key contributions:

- Built the entire frontend from scratch using React 19 + TypeScript + Vite 8 + React Router 7 + Tailwind CSS 4 + Zustand, including project structure, lazy-loaded routes, layout system (header/footer/content), and centralized route constants.
- Translated Figma PC/mobile designs into pixel-accurate pages: Landing (hero, services, animated stats), About (values, certificates, scroll-driven counters), Project (sidebar + mobile carousel for 6 case studies), Outsourcing, IT Solution (category sidebar / mobile pill tabs), and AI Matching.
- Implemented marketing UX details: sticky/blur page titles, service dropdown & mobile drawer navigation, IntersectionObserver scroll animations, animated metric counters, floating back-to-top, and Pretendard/Roboto typography aligned to design tokens.

Outcome: Delivered and continuously refining a production corporate website that presents Prototie's services and portfolio with Figma-faithful UI.

2. Protocare Partner Web — Feature 1 (Jun 2026 – Jun 30, 2026)

Team: Sole Frontend Developer, collaborating with Korean Design, Backend, and Project Management teams | **Role:** Frontend Developer / Frontend Owner

Overview: Role-based partner portal frontend for a care/support service platform, built from scratch from Figma PC/mobile designs, design system guidelines, and API notes/Excel specifications; covering Public, User, Support Staff, and Admin experiences with Korean/English multilingual support. Feature 1 delivery completed on Jun 30, 2026.

Key contributions:

- Built the entire frontend from scratch as the sole FE developer using React 19 + TypeScript + Vite, including project structure, routing, layout system, Tailwind CSS 4 tokens, reusable components, and responsive PC/mobile UI foundations.
- Translated Figma PC/mobile designs and design system assets into reusable, responsive UI components while aligning implementation with API notes/Excel specs and backend contract changes.
- Built support portal modules including dashboard, care search/detail, schedule calendar, settlement, profile, supporter application, notifications, announcements, FAQ, terms, and 1:1 inquiry flows.
- Implemented role-based API layers with Axios clients, cookie-based JWT authentication, refresh-token reissue interceptor, route guards, Zustand state, and TanStack Query server-state management.
- Collaborated frequently with Korean PM/design/backend teams through English messages to clarify Figma flows, API notes, UI feedback, and feature requirements.
- Set up and wrote unit tests with Vitest, React Testing Library, jsdom, and MSW across 54 test files, covering helpers, validation logic, Zustand stores, API services, hooks, reusable components, and key support screens.

Outcome: Delivered Feature 1 of a scalable, role-based partner web platform with production-oriented architecture, maintainable UI patterns, API-ready frontend modules, and a practical unit-test foundation (54 test files) for safer development.

3. Chatbot AI (Mar 2026 – Jun 2026; minor UI support ongoing)

Role: Frontend Developer / Chatbot UI Owner

Overview: Embeddable AI chatbot widget supporting real-time streaming conversations, chat history, rich UI cards, and interactive service flows directly inside a host website.

Key contributions:

- Built a reusable <dalya-chatbox> Web Component with Shadow DOM to isolate chatbot UI and styles from host pages.
- Integrated SSE / streaming chat APIs with Bearer token authentication, session handling, history list/detail APIs, loading states, DONE handling, and error handling.
- Implemented rich templates for vehicles, products, vouchers, stores, booking, pre-order/order completion, bill summaries, Q&A completion, YouTube previews, quick replies, and PII guard.
- Provided ongoing minor UI support and maintenance after the main implementation phase.

Outcome: Delivered an embeddable AI chat widget with responsive, isolated, and API-driven UI that can be reused across existing web pages.

4. E-commerce Platform (Client & Admin) (Jun 2025 – Feb 2026)

Team: 4 (2 Frontend, 2 Backend) | **Role:** Senior Frontend Developer / Frontend Owner

Overview: Full-featured e-commerce platform with client-facing storefront and admin management system. The platform went live in early Jan 2026 and was fully handed over in May 2026.

Key contributions:

- Designed frontend architecture from scratch using React/Next.js + TypeScript; enforced coding standards and reusable patterns.
- Implemented core flows: authentication, product listing, cart, checkout, order tracking, and account features.
- Built admin dashboard for user/product/order management with scalable table and form patterns.
- Integrated multiple online payment methods with robust error handling and secure request flows.
- Set up CI/CD pipelines with GitHub Actions and deployed to AWS (EC2/S3/Nginx).
- Optimized performance through memoization, efficient rendering, and bundle hygiene for production stability.

Outcome: Delivered and handed over a production e-commerce platform capable of handling real transactions with optimized performance and maintainable architecture.

5. Bingo Event Mini Game (Aug 2025 – Aug 2025)

Team size: 2 (1 Frontend, 1 Backend) | **Role:** Frontend Developer

Description: Interactive event-driven bingo mini game designed to increase user engagement and retention through gamified campaigns and real-time reward mechanics.

Responsibilities: Built end-to-end game UI including reward board, dynamic grid, and action flows; integrated APIs for game logic and reward processing; managed client-side state for instant feedback; and collaborated with backend to design efficient API contracts.

Results: Delivered a production-ready mini game that enhanced user engagement and established a reusable framework for future event-driven features.

6. Insurance Record Management System (Jun 2025 – Sep 2025)

Team: 5+ | **Role:** Senior Frontend Developer (Core team)

Overview: Enterprise system for managing insurance records and workflow processes.

Key contributions:

- Defined frontend architecture conventions and reviewed React best practices for maintainability.
- Supported feature implementation and improved performance in complex screens.
- Contributed to CI/CD practices to ensure reliable and frequent releases.

Outcome: Improved maintainability and ensured continuous delivery for a large enterprise codebase.

7. Water Park Ticketing System (Korea) — POS & Kiosk (Feb 2025 – Aug 2025)

Team: 1 | **Role:** Sole Frontend Developer

Overview: POS and Kiosk systems for on-site ticket sales across multiple water parks.

Key contributions:

- Built optimized checkout flows for high-traffic environments with resilience-first UI design.
- Integrated cash, bank transfer, and credit card payment workflows aligned with real-world operations.
- Configured domain setup and CI/CD pipelines for streamlined deployment.

Outcome: Delivered a stable production system handling on-site transactions in real environments.

PYRAMID SOFTWARE & CONSULTING — Software Engineer | Ho Chi Minh City, Vietnam

Jul 2020 – Jan 2025

1. Digital Notary System

Team: 4–6 | **Role:** Frontend Developer

Overview: Digital notarization platform serving both internal staff and external customers.

Key contributions:

- Developed internal and customer-facing web applications using React.js.
- Implemented document workflow screens and integrated REST APIs for end-to-end flows.

Outcome: Streamlined notarization processes and reduced manual handling.

2. One Gate System (Government Platform)

Team: 5+ | **Role:** Frontend Developer

Overview: Government service platform connecting citizens with state officials.

Key contributions:

- Developed React.js UI screens and integrated APIs for multi-user workflow modules.
- Ensured responsive, cross-device experience and stable UI behavior.

Outcome: Enabled efficient workflows across departments and users.

3. E-Office System (School & Provincial Versions)

Team: 6+ | **Role:** Frontend Developer

Overview: Document and task management systems for administrative organizations.

Key contributions:

- Designed responsive UI and implemented task/document workflows.
- Integrated APIs with ASP.NET backend services.

Outcome: Supported daily operations for schools and provincial offices.

PERSONAL PROJECT

Attendance & Payroll Management System — Full-stack Personal Project | Jan 2026 – Present (Active development)

Role: Full-stack Developer / Product Owner | **Team size:** 1

Tech stack: React 19, TypeScript, Vite 6, React Router 7, TanStack Query v5, Zustand, React Hook Form, Axios, i18next (EN/VI), Tailwind CSS 4, FullCalendar, html5-qrcode; Node.js, Express 5, Prisma 7 (+ pg adapter), PostgreSQL 16, JWT/bcrypt, cookie-parser, Zod, Swagger/OpenAPI, ExcelJS, WebSocket (ws), Docker Compose; deployed FE on Vercel.

Overview: Multi-portal HRM platform with separate Admin, HR, and Employee apps on one SPA — covering company/branch/position structure, staff membership (multi-branch, employment phase/type, salary), GPS/Wi-Fi attendance punch, leave/OT/late/early/issue approval workflows, real-time notifications, work schedules, holidays, timekeeping, and Vietnam-aware monthly payroll (settings, salary components, tax brackets, VN statutory caps, calculate → payslip → lock).

Key contributions:

- Designed end-to-end architecture: Prisma schema & migrations, REST API contracts (stable FE error/i18n codes), and three auth portals (admin / hr / employee) with separate login endpoints, refresh cookies, route guards, and role-based access rules (SUPER_ADMIN, STAFF, HR, WORKER).
- Built Admin web modules with reusable table/form patterns, pagination, keyword search, filters, detail/CRUD pages for companies, branches, positions, staff/members, accounts, attendance locations, attendance records (Excel export), work schedules, holidays, and payroll v1 screens (settings, components, tax brackets, VN statutory, timekeeping, approvals, payroll runs).
- Implemented mobile-first Employee & HR portals: GPS/Wi-Fi check-in/out with branch lock, month summary/history (including forgot-checkout status), workplace selector, QR-based HR invite + worker accept/reject, day attendance summary, and approval inbox (leave / OT / late / early / ISSUE with time override).
- Frontend data layer: TanStack Query + Zustand per portal, Axios with Credentials + 401 refresh-token retry, UTC calendar date handling, and EN/VI i18n synced with backend fe-api-codes.
- Backend modules: Express controllers/services/routes, Prisma ORM with soft-delete audit fields, auto employee-code generation, multi-branch membership, invitation flow, paginated queries, and Swagger-documented APIs.
- Real-time notifications via REST inbox + WebSocket (/ws) for company join confirm, member removed, and attendance request reviewed events; notification i18n keys for FE interpolation.
- Vietnam payroll engine: progressive PIT with family dependents deduction, BHXH base, VN allowance statutory revisions/caps, nationals-only components, late/early penalty modes, payroll calculate / results / lock, and Excel-friendly payslip snapshots.

Outcome: Ongoing full-stack product demonstrating ownership of multi-portal product design, secure authentication, real-time notifications, Vietnam payroll rules, database modeling, API design, and scalable React architecture — actively developed and deployed (FE on Vercel).

EDUCATION

Nguyen Tat Thanh University — Bachelor of Information Technology

2016 – 2020