
Dixon Minnick

Staff Frontend Engineer · AI-Assisted & Agentic Development

415.797.7609
Dixon@Minnick.dev

[LinkedIn](#)
[Github](#)

SUMMARY

Thoughtful and dedicated staff frontend engineer with over a decade of experience building and refactoring complex web applications in TypeScript, React, and Vue, as well as other legacy JavaScript frameworks. At Arqu, lead front-end engineer on a Vue 3 insurance-workflow platform and architect of the team's AI-assisted front-end development system building the custom agent skills, quality gates, and reviewer pipelines that let engineers and non-engineers alike ship production front-end safely. Equally at home going deep on UX, accessibility, and performance, or designing the tooling that multiplies a team's output.

SKILLS / LANGUAGES / TOOLS

- Primary Languages: TypeScript and JavaScript (ES6)
- Claude Code, Cursor, MCP, agentic workflows, custom skills & plugins, agent quality gates, prompt & rules engineering
- Frontend: React JS, NextJS, Redux, React Hooks, Vue 3 (Composition API, Pinia)
- CSS: Tailwind CSS (v4), shadcn, Styled Components, Emotion, SASS, PostCSS, Bootstrap
- Backend: NodeJS, Django and Python, ExpressJS, JSON Schema
- Others: Unit Testing, WCAG/ADA Accessibility, A/B Testing, Feature Flagging, API integration

EDUCATION

Tufts University, Medford MA - *BS Computer Science*

2011 - 2015

EXPERIENCE

arqu, Remote - *Staff Software Engineer*

JUNE 2024 - PRESENT

- Lead front-end engineer on "Doubtfire", a Vue 3 + TypeScript SPA (Vite, Pinia, Tailwind v4)— a custom AI-integrated inbox and deal-workspace client for the insurance broking platform
- Architected the team's AI-assisted development system — CLAUDE.md / AGENTS.md guidance plus custom Claude & Cursor skills (scaffolding and decomposing components, stores, and composables; convention-aware code review; theme and accessibility audits) that standardize how engineers and coding agents build across front-end repositories.
- Built automated quality gates for agent-authored PRs (complexity, file-size, dead-code, test-coverage, and style-budget checks with grep-able rule IDs) so both AI agents and non-AI contributors meet the same coding standards
- I own the Vue component libraries & design systems (shadcn-vue + Tailwind tokens) — built and unified core primitives (Button, Tooltip, accessible loading states, form inputs)

(arqu, continued)

- Enabled non-engineer product staff to ship production frontend: defined a guided branch-and-deploy workflow plus I directly reviewed their AI-assisted design documents, implementation specs, and 70+ pull requests (including the mobile-web implementation of the doubtfire email client)
- Built the rich-text email composer on TipTap (tables, images, links, font controls, sanitized paste, draft-first send) plus reusable email templates.
- Architected and implemented multi-step "Deal Kickoff" deal-submission wizards, V3 within the legacy platform and V4 within the doubtfire email client, consisting of form inputs with best practice form validation UX using zod.
- Re-encapsulated the Deal Manager dashboard as a drag-and-drop modular-widget workspace
- Built the Deal Conversion dashboard: a schema-driven UI highlighting top priority deals leveraging Chart.js visualizations and conditional rendering

Pre-Seed Stealth Startup, Remote - Software Engineer **SEP 2023 - NOV 2023 (Contract)**

- Using NextJS, TypeScript, React and TailwindCSS to create a webapp MVP leveraging AI

RxSense, Remote - Senior Software Engineer **MAY 2020 - JAN 2023**

- Refactored a white-label pharmacy benefit member experience for the web using vue.js integrated with a CMS for custom branding. This included a full redesign of an asynchronous multi-endpoint pharmacy pricing tool.
- Architected and implemented a full ground-up refactor of the white-label pharmacy benefit member experience using React, TypeScript, React Hooks, Styled Components, and material-ui.
- Migrated core front-end application logic from a 1-to-1-to-many relationship of login-to-benefit record-to-profiles to a 1-to-many-to-many relationship, along with the corresponding UI components and API interfaces.
- Implemented unit-testing and documentation strategies for the codebase using Jest, TS-Doc, and Storybook. This custom Storybook control to display custom-branding of white-label components, and mock application data test utils..
- Implemented a redesign of a pharmacy network pricing administration tool using vue.js
- Implemented a redesign of a conflicted record resolution UI within the admin tool using vue.js

Vector, San Francisco CA - Software Engineer **NOVEMBER 2019 - MAY 2020**

- Crafted schema-based user interfaces rendered with React
- Introduced and documented git workflow and best practices for the engineering team
- Architected and implemented an enterprise invoicing user interface and workflow

Chegg - Tutors, San Francisco CA - *Software Engineer*

AUGUST 2016 - JUNE 2019

- Architected & implemented migration of Tutors front end from backbone.js and Django to ES6, React & Redux using the [Open Components](#) strategy and developed a Django template tag to migrate components into the existing monolith
- Architected, prototyped and implemented chat-based tutoring client from its inception using TypeScript, React, Redux, Styled Components and a 3rd party messaging API
 - Worked on various features including messages, custom inputs, modal architecture
 - Researched and advocated for tech stack selections, including Formik, create-react-app, and TypeScript
 - Worked with Fabric.js to create a custom whiteboard tutoring solution for the client
- Contributed to the “Chegg Styled” cross-team initiative from its inception to migrate UI elements to a shared Styled Component library, built in React, to improve brand and UI consistency, as well as accessibility. I led the implementation of the library usage API and its documentation.
- Implemented front end migration of Tutors platform to Chegg-wide auth system
- Participated in Chegg Hackathons and Design Sprints to rapidly prototype potential features, including a tutor payment center and a chat-bot experience, parts of which were later adopted in features deployed to production
- Implemented A/B tests for multiple features including:
 - lesson scheduler calendaring experience to maximize tutor’s underutilized availability
 - lesson request form to collect information from students to facilitate better matching

Chatous - Kiwi, San Francisco CA - *Software Engineer* **JANUARY 2016 - APRIL 2016 (Contract)**

- Front-End Engineer on Kiwi.qa social media platform, developing features, and fixing bugs
- Migrated front-end from EJS and jQuery to React and Flux single page application
- Implemented various features including replies/comments to answers, hover profile cards, publicly answerable questions, nearby questions (using Google Maps API) and direct messages

AcquireMedia - NewsEdge, Burlington MA - *Software Dev Intern* **MAY 2014 - AUGUST 2015**

- Developer for NewsEdge platform, fixing bugs and adding features, working full time each summer and part-time during my senior year of college