Daniel Shaby

Full Stack Software Engineer

+1-818-370-8344 | dshaby@berkeley.edu | LinkedIn | GitHub | https://danielshaby.com

PROFESSIONAL SUMMARY

Full Stack Software Engineer with 4+ years of experience building production-grade systems across modern **React frontends** and **TypeScript/Node.js** & **Python backend services.** Experienced in designing **type-safe APIs**, **serverless microservices**, and scalable **UI architectures** deployed on **AWS** (Lambda, API Gateway, DynamoDB). Strong background in **CI/CD automation**, **Docker-based workflows**, and shipping quickly in startup environments. Proven track record evolving AI-enabled features from rapid prototype to production-grade systems. **UC Berkeley alum** focused on clean system design, strong contracts, and reliable engineering.

KEY SKILLS

Frontend: React, Next.js (App Router), TypeScript, JavaScript, Tailwind CSS, Material UI
Backend: Typescript (Node.js), Python (Flask, Django), AWS Lambda, Serverless Framework v4, Express, Nest.js
APIs: tRPC, REST, GraphQL
Databases: DynamoDB, MongoDB, PostgreSQL
DevOps/Cloud: AWS (Lambda, API Gateway, S3, CloudFront, CloudFormation, IAM, Cognito, WAF, CodePipeline, CodeBuild, SSM), AWS CDK, Docker, Kubernetes, GitHub Actions, Jenkins, Vercel
Testing: Jest, React Testing Library, Python Unittest, Zod validation
Tools: Redux (RTK) Query, Agile (Jira, Asana), Infrastructure as Code (IaC), Type-safe contracts.

PROFESSIONAL EXPERIENCE

Full Stack Engineer – *TS Industrial LLC* **March 2025—Present**

- Leading migration of a **9M+ product SKU** e-commerce platform to a **TypeScript-based serverless microservices architecture** on **AWS** (Lambda, API Gateway, DynamoDB).
- Developed and standardized backend service patterns using **Serverless v4**, **Zod validation**, and **shared contracts** across `products`, `users`, and `orders` microservices for consistent API design and maintainability.
- Designed and built backend utilities in Python and TypeScript for data transformation and internal automation workflows.
- Architected and deployed **Next.js 15 frontend** (App Router, RTK Query, Tailwind CSS), integrating with serverless backend services via shared type-safe contracts.
- Implemented **multi-environment CI/CD pipelines** (dev, test, UAT, prod) using AWS CodePipeline, CodeBuild, and GitHub integrations.
- Defined **infrastructure-as-code** with **AWS CDK** and custom Serverless configurations (WAF, CloudFront, OAC, S3 asset buckets, CloudWatch log retention).
- Introduced **type-safe shared** `@tsindustrial/contracts` for frontend + backend, improving **observability and standardizing error responses** (problem+json).
- **Optimized UI and search/order flows**, improving load time, search responsiveness, and component maintainability.
- Leading migration of the platform from tsiparts.net to modern-app.net, improving performance and long-term scalability.

Founding Full Stack Engineer – *Homekey - MVP Launch* **Jan 2025—Feb 2025**

- Launched a new product from scratch (zero code) to production, enabling 30+ users to buy/sell homes via the platform.
- Designed and evolved AI-powered search from prompt-based prototype to a scalable layered architecture using MongoDB filtering and vector search, improving performance and long-term maintainability.
- Developed a prioritized Minimum Viable Product (MVP) feature plan for release within one month, followed by building core features, including authentication and AI-powered PDF generation to support the launch.
- Integrated automatic emails and calendar scheduling tools to onboard and convert 20 waitlisted users into active testers.
- Deployed multi-environment infrastructure (dev, stage, prod) with DNS and AWS S3 for secure, shareable legal document storage and authentication via Clerk.
- Mentored 3 engineers on React, TypeScript, and Python Flask best practices, ensuring scalable and production-grade code.
- Led adoption of Redux Toolkit Query by designing a type-safe, scalable structure that improved readability, reduced boilerplate, and enabled a 6-person team to ship features more efficiently.
- Used early user feedback to guide architectural improvements and refine AI search performance.

Front-End Engineer – *Bynder* **Jan 2023—Feb 2024**

- Developed micro-frontends enabling CSV-based asset migration from Webdam to Bynder, supporting thousands of enterprise users with real-time error resolution.
- Advocated for and launched the first Next.js-based project in the organization by developing an internal case-of-support.
- Implemented **Docker/Kubernetes**-based CI/CD pipelines, improving deployment consistency and developer velocity.
- Built internal analytics dashboard integrating frontend UI with backend asset processing services and performance logging systems.
- Modernized image crop tool by replacing legacy jQuery/ColdFusion with a React-based solution, enhancing maintainability and UX.
- Tested new features end-to-end with consistent unit/integration tests using React Testing Library and Jest.
- Implemented skeleton loaders and caching strategies, improving page load times by 30%.

Full Stack Software Engineer – *Crown Clothing* **Mar 2022—Dec 2022**

- Redesigned checkout flow and React UI, increasing online sales by 15%.
- Migrated state management to Redux for maintainability.
- Integrated Apollo Client with GraphQL APIs and optimized frontend assets, pagination, and lazy loading strategies, boosting Lighthouse score from 77% to 93%.
- Built secure OAuth 2.0 authentication to improve UX and user security.
- Refactored high-traffic API endpoints to improve reliability and performance.

Cofounder & Product Team Lead – *BOTE* **Apr 2020–Feb 2022**

- Led cross-functional team of 8+ developers, designers, and content creators from USC, UCLA, and Chapman to deliver product MVP.
- Built MVP using React, Node.js, Bootstrap, and Stripe for payments.
- Designed and deployed marketing pages and landing experiences, increasing site traffic to 10K monthly views and 600 monthly users.

PROJECTS

Math Photo Solver (Next.js, tRPC, TypeScript, Zod) | Live Demo **Jan 2025**

- Built full-stack OCR-based math solver using Next.js and tRPC, returning structured step-by-step explanations via Wolfram Alpha APIs.
- Designed type-safe APIs using tRPC and Zod to enforce consistent client-server contracts.
- Implemented environment validation and error handling to prevent runtime configuration failures.

Daniel's Blog (Django, PostgreSQL, AWS S3) **June 2024**

Built full-stack blog (Django, PostgreSQL, AWS S3), deployed via Elastic Beanstalk.

Real-Estate (NextJS, Typescript, OpenAI, Redfin API) **May 2024**

Built real estate platform with LLM-enhanced search and Google APIs; implemented global state with RTK Query.

EDUCATION

University of California, Berkeley
Bachelor of Arts in Economics & Public Health (Double major) Graduated 2020 | GPA: 3.7

CERTIFICATIONS

LinkedIn Learning/Udemy: Sr. React Developer, Docker & Kubernetes, System Design **2020–2024**