HARSH DUBEY

Frontend Developer | Full Stack AI

Dwarka, Delhi | +91-6387550457 | harshdazz4@gmail.com

linkedin.com/in/harsh-dubey-b2b2b4395 | github.com/harshdazz | heyharshdubey.vercel.app

PROFESSIONAL SUMMARY

Frontend Developer with 1 year of professional experience who worked across the full stack of a production AI stock research platform. Built responsive **React** and **Next.js** interfaces, shipped **RESTful APIs** in **Python (FastAPI)** and **Node.js**, and integrated **AWS** services, **Redis** caching, **Docker** deployments and LLM features such as **RAG** chat and real-time streaming. Writes **unit and integration tests** for every feature, ships through **CI/CD** pipelines, and works in Agile sprints with product managers, designers and engineers.

PROFESSIONAL EXPERIENCE

BNCW ENTERPRISES | Frontend Developer **September 2025 to September 2026**

Worked across the Next.js frontend, FastAPI backend and RAG chatbot service of an AI stock research platform: 200+ Jira tickets, 400+ merged PRs.

- **Front-End Development:** Built responsive, reusable **React** UI with **Redux Toolkit**, **Tailwind CSS** and **Material UI**, including a rich text editor with video embeds and drag-and-drop uploads, and **JWT** login, password reset and email verification flows that let users sign up and recover accounts without support help.
- **RESTful APIs and Backend Services:** Designed and shipped **FastAPI** endpoints with **SQLAlchemy**, **Pydantic** validation and **Alembic** migrations (content reordering, subscription billing status, permission guards), and **Next.js** API routes that proxy market-data APIs, keeping provider keys out of the browser.
- **Cloud Integrations:** Integrated **AWS S3** media uploads and **AWS SES** transactional email, and moved community bans and admin-only actions into server-side checks so moderation rules cannot be bypassed from the client.
- **Data Pipeline Service:** Extended the **Python** ingestion service of the RAG chatbot with a typed source contract and per-source backfill, so each content source syncs independently, one failing source no longer blocks the rest, and new sources are easier to add; tested with **pytest**.
- **Real-Time Streaming:** Implemented **SSE**-based token streaming for the AI research assistant, so answers render as they are generated instead of after the full response, and a sanitized markdown path that strips script injection (**XSS**) from model output.
- **Debugging and Quality:** Fixed intraday 1D and 5D stock charts to show correct trading-hour data by anchoring them to US market sessions, built interactive **Recharts** dashboards, and added **Vitest** tests and **PostHog** tracking to catch regressions and measure feature usage.

PROJECTS

RouteLens | TypeScript, Node.js, VS Code Extension API, Model Context Protocol, Zod, Vitest, GitHub Actions

- Published an open source **VS Code extension** (VS Code Marketplace and Open VSX) that discovers every **Next.js** and **Express** API route with the **TypeScript compiler API**, without running user code, and generates request bodies that satisfy each route's **Zod** schema.
- Added one-click testing of all routes, multi-step flows and keychain-backed auto-login, plus an **npm CLI** for **GitHub Actions**.
- Shipped an **MCP server** exposing 7 tools that let **AI agents** such as Claude Code and GitHub Copilot run the API, read a failing route's source and fix it within host guard rails; backed by about 380 unit and integration tests, 17 end-to-end tests and an approval-gated **CI/CD** release.

AI Mock Interview Platform | React 19, TypeScript, Google Gemini, Cloudflare Workers, Firebase, Zod, Web Speech API

- Built a mock interview app that reads the candidate's **PDF** CV natively with **Google Gemini** to write role-specific questions (4 of 5 cite concrete CV claims in testing), transcribes spoken answers with live **speech to text**, and asks a follow-up on the biggest gap.
- Designed evidence-based grading: **structured outputs** return a covered or missed verdict with a transcript quote per rubric point, code verifies each quote and computes the score, and a 13-case **golden-set LLM evaluation**, including a **prompt injection** case, passes 13 of 13.
- Kept the API key off the client with a **Cloudflare Workers** AI gateway that verifies **Firebase** ID tokens, rate-limits each user to 30 requests per hour with **Durable Objects**, validates input with Zod, and falls back across Gemini models on quota and overload errors.

ACHIEVEMENTS AND CERTIFICATIONS

- **AWS Educate Introduction to Generative AI**, Amazon Web Services Training and Certification (Credly badge)
- **Open Source Contributor on GitHub:** contribute bug fixes and tests to open source AI and developer-tool projects

EDUCATION

United College of Engineering and Research, Prayagraj | Bachelor of Technology in Computer Science **2021 to 2025**

TECHNICAL SKILLS

Languages: JavaScript (ES6+), TypeScript, Python, HTML5, CSS3, SQL

Front-End: React.js, Next.js, Redux Toolkit, Tailwind CSS, Material UI, Recharts, Responsive Design

Back-End: Node.js, Express.js, FastAPI, RESTful APIs, Server-Sent Events, SQLAlchemy, Pydantic, Zod, JWT Auth, Webhooks

Databases: MongoDB, Firestore (NoSQL), PostgreSQL, MySQL, pgvector, Redis (Caching)

Cloud and DevOps: AWS (S3, SES, RDS), Serverless (Cloudflare Workers, Firebase), Docker, Deployment, CI/CD, GitHub Actions, Git, Sentry, Jira

AI: Generative AI, LLM Integration, LLM Evaluation, LLM Observability (Langfuse), RAG, Structured Outputs, MCP, OpenAI, Claude, Gemini