

Dharun R

Full Stack Developer

+91 6383678731 dharunamikaze@gmail.com [linkedin.com/in/dharunn](https://www.linkedin.com/in/dharunn) github.com/DharuNamikaze [Portfolio](#)

Education

KIT - KALAIgnARKARUNANIDHI INSTITUTE OF TECHNOLOGY Nov 2022 – April 2026
Bachelor of Artificial Intelligence and Data Science (CGPA: 8 / 10) Coimbatore, Tamil Nadu.

- **Relevant Coursework:** Data Science, Machine Learning, Data Structures and Algorithms.

Technical Skills

- **Core:** TypeScript, Node.js, React, Python, Next.js
- **Web:** HTML, CSS, Tailwind CSS, Express, Astro.js
- **Databases:** MongoDB, Firebase, Postgres
- **Tools:** Git, GitHub, Docker, Kubernetes, RAG, MCP
- **ML:** Numpy, Sklearn, Pytorch
- **Concepts:** Machine learning, Database, Operating system, System design

Experience

MyAnatomy Integration June 2025 - Dec 2025
FullStack Developer (Remote) Coimbatore, Tamil Nadu.

- Contributed to AgentHub – an AI-driven support platform enabling clients to build RAG-powered assistants with real-time chat, human takeover, and analytics.
- Built document-ingestion and retrieval pipeline using OpenAI Assistant + Pinecone **RAG**, enabling fast, context-rich responses for support agents using **React** and **Node**.
- Implemented a **WebSocket**-based live chat with seamless human-takeover, preserving conversation context and enabling agent escalation without user friction.
- Developed dashboard backed by **MongoDB** and auto-refreshing analytics APIs to surface real-time KPIs for teams.
- Optimized frontend build tree-shaking, code-splitting, lazy-loading, reducing bundle size by **39%**.
- Delivered test coverage: **529** backend tests, **128** frontend tests, plus property tests **85%** overall coverage.

Projects

Mithilla | *Next.js, TypeScript, Medusa.js, NeonDB-Postgres, Razorpay (Client Project)* Mithilla.vercel.app

- Engineered a full-stack e-commerce storefront with **Next.js 16** and Medusa v2 headless architecture – decoupling the **React** frontend from commerce logic for independent scaling across Vercel and Render.
- Adopted Next.js 16 Partial Prerendering and **use cache** with tag-based invalidation – pages serve from cache on every visit yet reflect backend updates within seconds.
- Secured auth with HttpOnly cookies, server-side input validation, and Upstash **Redis** sliding-window rate limiting across 5 API surfaces – blocking brute-force attacks without adding login friction.
- Integrated **Razorpay** as a pluggable Medusa payment module and built a custom Cloudinary file provider for **CDN** image delivery – replacing two default services with production-grade alternatives.
- Architected for scale with distributed Redis **rate limiting** across serverless instances, server-side product filtering at the Medusa query layer, and an idempotent seed workflow – reducing infrastructure overhead and enabling new market onboarding without code changes.

Gravity | *TypeScript* github.com/DharuNamikaze/Gravity

- Gravity enables LLMs to inspect and debug live UI by querying real DOM values, unlocking UI debugging capabilities not possible with static DOM snapshots.
- Built a Manifest V3 chrome extension that invokes the chrome.debugger API (CDP) to extract runtime DOM, box models, and computed styles from active tabs.
- Implemented a end-to-end pipeline (LLM ↔ **MCP** ↔ Native Host ↔ Extension ↔ Browser) with automatic reconnection.
- Packaged the system as an **npm package** which works with almost all MCP clients (VS Code, Cursor, etc.).