YASH KEDIA

Full Stack Developer

Hyderabad, India | yashkd12790@gmail.com | kediayash.in | github.com/spiderkd

Summary

Full stack developer specializing in Next.js and TypeScript, building production web applications across e-commerce, fintech, and SaaS domains. Experienced with AI-integrated tooling (Vercel AI SDK, MCP), cloud infrastructure (AWS), and modern auth/data layers (BetterAuth, Prisma, Postgres/MongoDB).

Experience

Nextlayer LLP

2024 – 2026

Software Developer · Digital agency delivering client projects across e-commerce, fintech, and internal tooling

- Shipped 8+ production web applications over 2 years across e-commerce, fintech, and personal management systems using Next.js, Node/Express, MongoDB, and PostgreSQL
- Architected REST APIs and database schemas supporting 5,000+ active users per deployment, cutting average API response time by ˜30% through query optimization and indexing
- Migrated 3 legacy client applications to Next.js/TypeScript, reducing production bugs by ˜40% and improving Lighthouse performance scores by 20+ points
- Managed 2–3c parallel client projects simultaneously, owning requirements scoping through deployment

Freelance

2025 – 2026

Full Stack Developer

- Built a personal management system for Vitaan using AWS S3 storage and role-based access control across 3 user roles, cutting manual admin work by an estimated 15+ hours/month
- Developed the marketing website for Matcha Studio (design agency) using Next.js and Sanity CMS, cutting client content-update turnaround from days to minutes
- Delivered portfolio websites for 4+ independent clients end-to-end, from design to deployment

Mentorsity

2023 – 2024

Software Developer Intern · Ed-tech platform connecting university students with mentors

- Rebuilt core frontend interfaces in React/Next.js, improving page load performance by ˜25%
- Built and integrated 10+ backend API endpoints in Node/Express powering student-mentor matching and profile features
- Shipped 6 features over a 10-month internship in an agile, cross-functional team of designers and engineers

Projects

By Default Human

bydefaulthuman.fun | github

Open-source, MIT-licensed React component library offering 50+ hand-drawn UI components and layout blocks. Every border, line, and shape is rendered live at runtime through Rough.js rather than static SVG assets, giving each component an authentic sketched look that redraws itself on mount. Supports three switchable stroke themes (pencil, ink, crayon) toggled globally via a provider or per-component, and installs through a shadcn-style CLI so consumers copy the source directly into their own repo instead of pulling an opaque npm dependency, keeping the library fully customizable and lock-in free.

- 50+ components and blocks spanning charts, cards, sticky notes, and interactive primitives
- Runtime SVG generation via Rough.js — zero static image assets shipped

Next.js, TypeScript, Rough.js, shadcn CLI

BrainDump

braindump-livid.vercel.app | *github*

AI agent that turns unstructured, stream-of-consciousness input — half-sentences, voice-memo transcripts, 2am thoughts — into clean, organized Notion pages with zero manual sorting. A custom MCP toolchain reads through the raw dump, extracts the actual tasks and ideas buried in it, then titles, tags, and structures the content before filing it straight into the user's connected Notion workspace via the Notion MCP, returning a direct link to the finished page.

- Custom Express-based MCP server orchestrating parsing, structuring, and Notion page creation end-to-end
- Chained tool-calling pipeline built on the Vercel AI SDK to coordinate multi-step agent actions

Next.js, Vercel AI SDK, custom Express MCP backend, Notion MCP

HLS Video Transcoder

video-transcoder-main.vercel.app | *github*

Cloud video-processing pipeline that converts uploaded video into adaptive-bitrate streams. Uploaded files land in AWS S3 and a job message is pushed to AWS SQS, which is polled to trigger a dynamically spun-up Docker container running FFmpeg. The container transcodes the source video into 3 separate resolutions packaged as HLS (.m3u8) playlists, then writes the output to a dedicated bucket ready for adaptive streaming playback in the browser.

- Queue-driven, containerized transcoding workers spun up on demand via Docker
- Multi-resolution adaptive bitrate output (HLS/.m3u8) for smooth playback across network conditions

React, Docker, FFmpeg, AWS S3, AWS SQS

Tech Stack

Languages:	JavaScript, TypeScript, Python
Frontend:	React, Next.js, Tailwind, GSAP
Backend:	Node.js, Express, Prisma, BetterAuth
Databases:	MongoDB, PostgreSQL, Supabase
Cloud & DevOps:	AWS, Docker, Vercel, Git

Education

BIT Sindri, Dhanbad, India

2022 – 2026

B.Tech, production & Industrial