



Momin Shaikh

Full-Stack Software Engineer | AI Integration Specialist

+880-1405-374822 | thisismominshaikh@gmail.com

linkedin.com/in/themominshaikh | github.com/mominshaikhdev | mominshaikhdev.vercel.app

PROFESSIONAL SUMMARY

Full-stack software engineer with hands-on experience building production-grade web applications using React, Next.js, Express.js, Django, Laravel, and AI integrations. Proven ability to design scalable architectures including multi-step agentic AI systems, multi-tenant SaaS platforms, RAG-powered AI platforms, and real-time applications. Strong foundation in data structures, algorithms, and system design. Fluent in English and Bengali.

TECHNICAL SKILLS

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

Frameworks & Libraries: React.js, Next.js, Express.js, Node.js, Django, DRF, Laravel, LangChain, LangGraph, Zustand, TanStack Query, Tailwind CSS, REST, GraphQL, WebSockets, Stripe API, OpenAI API, Gemini API, Groq SDK, Better Auth, Clerk, Sanctum, GSAP, Framer Motion, Three.js, Recharts

AI & ML: Retrieval-Augmented Generation (RAG), OpenAI GPT, Google Gemini (Function Calling / Tool Loops), Groq (Llama 3.3 70B, Llama 3.2 Vision), Agentic AI Systems, Whisper STT, TTS, Vercel AI SDK, Vector Databases (Pinecone/Qdrant), MLOps

Databases: PostgreSQL, MySQL, MongoDB, Redis, Prisma ORM, Mongoose ORM, Eloquent ORM, query optimization, N+1 elimination, eager loading

Tools & Platforms: Git, GitHub, Docker, Kubernetes, GitHub Actions (CI/CD), AWS, GCP, Vercel, Railway, Sentry, Linux/Unix CLI, Uploadthing, Vite, i18Next

Concepts: Multi-tenant Architecture, RBAC, Agentic AI Systems, Dual-DB Architecture, Event-Driven Architecture, Microservices, SPA, CI/CD, RESTful API Design, Webhooks

EXPERIENCES

Independent | Open-Source Developer

Remote | 2020 - Present

Designed and shipped multiple full-stack open-source projects spanning agentic AI systems, SaaS platforms, real-time collaboration tools, and multi-tenant architectures - entirely self-directed.

- Built CodeFusion Research Agent - a Django/Next.js AI agent using Google Gemini 2.0 Flash function calling to autonomously explore GitHub repositories and answer deep technical questions.
- Engineered Collaborative Team Hub with dual-database architecture (PostgreSQL + MongoDB), real-time Socket.IO, JWT auth, RBAC, and Kanban workflows deployed on Railway.
- Contributed reusable, well-documented codebases to GitHub; maintained production deployments on Vercel and Railway with CI/CD pipelines.

WebFlow Developer

Flow Seek - Khulna, Bangladesh | 2020 - 2023

- Designed and built 20+ responsive marketing websites and landing pages in Webflow for clients across e-commerce, real estate, and local service industries, translating Figma designs into pixel-perfect, production-ready sites while eliminating custom backend overhead and cutting delivery time.
- Architected reusable Webflow CMS collections (blog, portfolio, case studies, team profiles) that empowered non-technical clients to self-manage content post-launch, reducing recurring developer support requests and client turnaround time on routine updates.
- Implemented custom scroll-triggered animations and micro-interactions using Webflow's native interaction engine, improving on-page engagement and reducing bounce rate across delivered client sites.
- Optimized on-page SEO fundamentals - semantic HTML, meta tags, structured data, image compression, and Core Web Vitals - on every project, consistently achieving 90+ Lighthouse scores and improving clients' organic search visibility.

- Extended Webflow's native functionality with custom JavaScript embeds and third-party integrations (Google Analytics, Mailchimp, Stripe payment links, Zapier automations) to meet client-specific business and marketing requirements.
- Owned the full client engagement lifecycle - requirements gathering, wireframing, design handoff, staging review, and launch - coordinating directly with designers and stakeholders to deliver projects on time and within scope over a 3-year period.

PROJECTS

VentureForge AI | [Frontend](#) | [Backend](#) | [GitHub](#)

Next.js 15, Express.js, Node.js, MongoDB, Mongoose, Groq SDK (Llama 3.3 70B & 3.2 Vision), Better Auth, TanStack Query, Recharts, Tailwind CSS v4, Vercel | 2026

- Architected a full-stack AI startup incubator & financial validation portal connecting a Next.js 15 App Router frontend with an Express/MongoDB backend, deployed cross-origin on Vercel with Better Auth session management.
- Engineered an agentic startup evaluation engine using Groq LLM (Llama-3.3-70b-versatile) to auto-score venture viability, generate structured SWOT analysis, recommend tech stack components, and perform competitor gap analysis.
- Built an interactive financial forecasting dashboard with CFO AI audit, ingesting CSV/manual financial ledgers to construct dynamic Recharts (Area charts for cash flow, Bar charts for net margins) and automated executive financial summary reports.
- Designed VentureMentor AI chat drawer, a conversational coaching drawer grounded in venture details that maintains session history in MongoDB and generates real-time follow-up prompt suggestions.
- Integrated multimodal pitch & image intelligence, using Llama 3.2 11B Vision for base64 image layout audits and document text parsing for instant pitch deck digests and risk summaries.

CodeFusion Research Agent | [Frontend](#) | [Backend](#) | [GitHub](#)

Django, Django REST Framework, PostgreSQL, Google Gemini 2.0 Flash, Next.js 16, React 19, TypeScript, Tailwind CSS, Gunicorn, Railway, Vercel | 2026

- Architected a full-stack AI research agent that answers deep technical questions about any GitHub repository by autonomously cloning the codebase and exploring it through a bounded multi-step tool-calling loop powered by Google Gemini 2.0 Flash's native function calling - eliminating hallucination by grounding every answer in actual source files.
- Designed a custom agentic execution engine (`gemini_runner.py`) with a hard-capped 24-round tool loop using Gemini's FunctionDeclaration + FunctionResponse pattern; dispatches 7 tools (`list_files`, `read_file`, `search_code`, `get_file_summary`, `save_finding`, `get_previous_findings`, `list_past_sessions`), feeds results back as structured parts, and terminates cleanly on text-only response or marks session FAILED on round cap or exception.
- Engineered a 4-table PostgreSQL schema: Finding rows persisted incrementally so knowledge survives across sessions; references stored as denormalized JSONField to avoid a join hot-path on every session read; ToolCallLog written before results are fed back to the model, ensuring a crash-safe, pre-execution audit trail.
- Implemented context window management with configurable hard caps - 120 KB per file read, 40 max search results, 2,000-char per-finding truncation in LLM context - and a `get_file_summary` cheap-outline tool to avoid full reads, bounding token costs on large codebases.
- Built prior-knowledge surfacing: at session start the agent pre-loads past findings and session summaries into the initial prompt, saving 1-2 tool round-trips on repeat queries and enabling the model to build accumulated knowledge across independent sessions.
- Deployed Django backend on Railway (Gunicorn + psycpg2) with environment-driven SQLite to PostgreSQL switching, CORS-locked origins, and DRF JSON-only renderer; Next.js 16 / React 19 frontend on Vercel features an AgenticLoader visualising live tool-call progress and a full ToolCallLog audit view.

Collaborative Team Hub | [Live](#) | [GitHub](#)

Next.js, Node.js/Express, PostgreSQL, MongoDB, Prisma ORM, Socket.IO, Cloudinary, Nodemailer, Zustand | 2026

- Engineered a full-stack collaborative workspace platform with a Node.js/Express REST API backed by a dual-database architecture (PostgreSQL + MongoDB) abstracted through Prisma ORM, with an automated

deployment script that conditionally bootstraps either provider.

- Designed a stateless JWT authentication system with short-lived access tokens and database-persisted refresh tokens via HttpOnly cookies; the Axios client implements a transparent refresh interceptor with a shared promise queue to coalesce concurrent 401 failures and avoid redundant refresh calls.
- Implemented a layered middleware pipeline with configurable CORS, Helmet security headers, rate limiting, JWT authentication guard, workspace-membership gate, and an admin role enforcer - each composable and independently applied per route.
- Built a real-time Socket.IO layer with handshake-level JWT verification, workspace-scoped rooms, live typing indicators, and server-push events for all CRUD mutations, keeping all connected clients in sync without polling.
- Developed nine REST controllers covering auth, tokenized email invitations via Nodemailer, granular RBAC with a per-workspace JSON permission matrix, goal and milestone lifecycle tracking, Kanban action items with bulk Prisma-transaction reordering, emoji reactions, threaded comments with @mention notifications, a parallel-aggregation analytics dashboard with CSV export, and a cursor-paginated activity feed.
- Built the Next.js App Router frontend with a Zustand auth store, drag-and-drop Kanban board, command palette, optimistic UI updates, a permission-aware PermissionGate component, Cloudinary avatar upload, and workspace theming; deployed via Turborepo monorepo on Railway.

Bookified | [Live](#) | [GitHub](#)

Next.js, LangChain, OpenAI, Stripe, Clerk, Vercel AI SDK | 2026

- Engineered a production SaaS platform enabling users to chat with uploaded PDFs using a Retrieval-Augmented Generation (RAG) pipeline built with LangChain and OpenAI, delivering contextually accurate answers in real time.
- Integrated bi-directional voice features via OpenAI Whisper (speech-to-text) and OpenAI TTS, enabling a fully conversational hands-free document interaction experience.
- Implemented real-time response streaming with the Vercel AI SDK, delivering word-by-word AI output to significantly reduce perceived latency and boost user engagement.
- Built a full SaaS billing and access-control system using Stripe (tiered subscriptions) and Clerk (auth), with custom middleware gating premium features such as voice and larger file limits behind a Pro plan.
- Developed a scalable document processing pipeline with Uploadthing and PDF-parse for secure cloud upload, text extraction, and vector indexing for AI consumption.

Multi-School Management System | [Live](#) | [GitHub](#)

Laravel, React, TypeScript, Vite | 2026

- Architected a multi-tenant database system supporting isolated data environments for concurrent schools using custom Laravel middleware and global query scopes, ensuring strict data privacy per tenant.
- Implemented a granular RBAC system with Laravel Gates and Policies covering four roles (Admin, Teacher, Parent, Student), with dynamic UI rendering based on access level.
- Engineered a dynamic GPA calculation engine adapting to diverse school-specific grading scales with automated PDF report card generation, translating complex business logic into efficient backend code.
- Optimized RESTful APIs using Laravel API Resources and resolved N+1 query problems via eager loading, significantly reducing database load and improving API response times.
- Maintained end-to-end type safety across the full stack using TypeScript, eliminating runtime errors in client-server data handling.

Acme Store | [Live](#) | [GitHub](#)

Next.js, Stripe, Tailwind CSS, Zustand | 2026

- Integrated Stripe Checkout Sessions for secure, professional-grade payment flows with automated order confirmation via Stripe Webhooks syncing the database with payment events.
- Built an SEO-friendly product storefront using Next.js Server Components with near-instant load times and persistent shopping cart via local storage using Zustand.

EDUCATION

LL.B (Hons) in Al-Fiqh and Law

Faculty of Law, Islamic University, Kushtia-Jhenaidah | 2019 - 2023 | CGPA: 3.28 / 4.00

Higher Secondary Certificate (H.S.C), Science

Nachole Govt. College, Rajshahi Board | 2018 | GPA: 4.58 / 5.00

Secondary School Certificate (S.S.C), Science

Maktapur High School, Rajshahi Board | 2015 | GPA: 5.00 / 5.00

ADDITIONAL

Languages: English (Fluent), Bengali (Native), Hindi (Working Knowledge)

Other Skills: Strong analytical background in Philosophy and Physics; experienced with terminal/CLI on both Windows and Linux.