

Sunil Kumar Mishra

Bangalore | sunil.crud@gmail.com | 63660 02870 | 96923 58853 | LinkedIn | GitHub

Profile Summary

Full Stack Developer with 3+ years of experience building production applications using Node.js, TypeScript, React.js, Next.js, and NestJS. Experienced in developing scalable REST APIs, real-time systems with Socket.io, asynchronous processing with Kafka, and data-intensive applications using MongoDB, PostgreSQL, and Redis. Built real-time applications supporting 1M+ concurrent users and deployed production workloads on AWS.

Skills

- **Languages:** JavaScript | TypeScript | HTML | CSS
- **Frontend:** React.js | Next.js | Redux | Zustand | React Query | Axios | Socket.io
- **Backend:** Node.js | Express.js | NestJS | REST | GraphQL | WebSockets | Kafka | SSE | gRPC
- **Databases:** MongoDB | PostgreSQL | Redis | DynamoDB | Elasticsearch | Qdrant | Mongoose | TypeORM | Prisma
- **Cloud/DevOps/AI:** AWS (EC2, S3, CloudFront, IAM, API Gateway, Serverless) | Docker | Kubernetes | CI/CD | Jest | Ollama | RAG

Experience

Nodejs Developer (Fullstack), Webskitters Technology, Kolkata Aug 2024 – June 2026

- Built a **real-time fantasy football platform** supporting **1M+ concurrent users** using **Node.js**, **React.js**, and **Socket.io**; implemented **live player trading**, **in-game chat**, **snake/ladder draft mode**, and **subscription-based purchases**. Leveraged **Kafka** for asynchronous event processing and horizontal scalability, reducing **live data latency by ~40%** during peak gameplay.
- Built a **scalable event management system** using **NestJS** and **React.js**; developed modular APIs for **event creation**, **ticket booking**, and **real-time notifications**. Implemented a media migration pipeline to **AWS S3**, **Redis**-based caching and session management, and used **PostgreSQL** for transactional **user**, **event**, and **ticket** management.
- Developed a **full-stack job portal** with **role-based access control (Admin, Candidate, Company)** using **Node.js** and **React.js**; used **GraphQL** to serve detailed job data to authenticated users while exposing a **limited public field set** to unauthenticated visitors, and integrated **subscription-based job posting**, **payment processing**, and **Elasticsearch** for fast search and filtering across **10,000+ job listings**. Built an **admin dashboard** for real-time monitoring of users, job postings, and billing.

Full Stack Developer, Searchingyard, Bhubaneswar Jan 2023 – Apr 2024

- Developed a **modular HRMS platform** using **React.js**, **Zustand**, and **React Hook Form/Formik**; built reusable components and drag-and-drop interfaces for **employee management**, **payroll**, and **attendance tracking**, supporting **50+ employees**.
- Developed a **full-featured Blog CMS** with admin workflows for **content approval**, editing, and moderation; leveraged **GraphQL** for flexible data fetching, **Redis** for caching frequently accessed posts, and **Elasticsearch** for fast full-text search, improving **content publishing speed by 60%**.
- Built a **canteen management system** integrated with **IOT-based card payment devices**; implemented **real-time order tracking**, **role-based access control (RBAC)**, and **Socket.io**-based one-to-one live chat for admins, canteen managers, and customers.

Breakdown Engineer, Swastik Engineering Aug 2021 – Jan 2023

Personal Projects

EduQuiz Platform — Full-Stack EdTech App (*in progress*) 2025 – Present

- Built a real-time quiz platform for Grades and Teachers using NestJS with a seamless architecture switch between **serverless** and **monolithic** deployments — enabling cost-efficient scaling without code changes.
- Admin panel in Next.js with TipTap rich-text editor for quiz authoring, TanStack Table for paginated listings, and granular feature toggles to enable/disable quiz, video, and notes per grade.

Local RAG Chatbot — Node.js + Qdrant + Ollama 2026

github.com/sunil380875/nodejs-rag-chatbot

- Built a fully local Retrieval-Augmented Generation API using Node.js, Qdrant vector DB, and Ollama (nomic-embed-text for embeddings, llama3 for generation) — zero external API cost, fully private inference.
- Implemented a custom text chunker with configurable overlap to preserve semantic context across chunk boundaries; designed `/ingest` and `/ask` REST endpoints with centralized error handling and standardized `ApiResponse` wrapper.
- Planned integration into EduQuiz as the syllabus-aware AI doubt-solving backend, replacing single-model inference with Qdrant-powered context retrieval for more accurate, grounded answers.

Education

Veer Surendra Sai University of Technology — B.Tech in Electrical Engineering 2018