

ROHIT SINGH

+91-9068703532 | rohitsinghwork11@gmail.com | [LinkedIn](#) | [Github](#) | [Coding](#)

SUMMARY

CS sophomore at KL University (CGPA 9.4/10) with hands-on experience shipping production-grade distributed systems, scalable REST APIs, and real-time event-driven platforms using Java, C++, Python, and TypeScript. Codeforces Expert · CodeChef 5★ · LeetCode Knight Top 5% (500+ solved). Seeking an SDE internship to architect scalable, high-impact backend systems.

EDUCATION

KL University, Vijayawada — B.Tech Computer Science & Engineering | CGPA: 9.42 / 10.0 Aug 2023 – Jul 2027 (Expected)

TECHNICAL SKILLS

Languages: Java · C++ · Python · JavaScript · TypeScript

Backend: Spring Boot 3.2 · Node.js · Express.js · Next.js 14 · Spring WebFlux · Socket.IO · REST API Design

Databases: MySQL 8.0 · PostgreSQL · MongoDB Atlas · JPA/Hibernate · 2dsphere Indexing · Query Optimization

DevOps & Cloud: Docker · Kubernetes (HPA) · GitHub Actions · Jenkins · Ansible · Prometheus · Grafana · Vercel · Render

Core CS: Distributed Systems · Event-Driven Architecture · OOP · System Design · JWT · RBAC · Multithreading · DSA

PROJECTS

FitLife Pro — AI-Powered Fitness Platform | [React](#) · [Spring Boot 3.2](#) · [MySQL](#) · [Gemini AI](#) · [Docker](#) · [Kubernetes](#) · [Jenkins](#) · [GitHub Actions](#)
[25Rohit25/FitLife](#)

- **What & Uniqueness:** Architected a full-stack **AI fitness platform (React/Vite · Spring Boot 3.2 · MySQL 8.0 · JPA/Hibernate)** uniquely combining a real-time **Google Gemini AI** coaching engine, an interactive analytics dashboard (**Recharts**), and a **gamification system** (XP · badges · streaks) — far beyond conventional fitness CRUD apps — containerized with **Docker Compose** and orchestrated on **Kubernetes** with **HPA** for auto-scaling.
- **Why It Exists:** Personalized fitness coaching is **cost-prohibitive** for most users, yet generic apps deliver **zero adaptive guidance**; this platform validates that gap by serving **live, context-aware** workout and nutrition plans through a secured **JWT + RBAC Spring Security** layer accessible to any authenticated user.
- **How It Solves:** Integrates **gemini-2.5-flash-lite** via **Spring Boot REST APIs** to generate dynamic coaching responses; dual **CI/CD pipelines (GitHub Actions + Jenkins Declarative)** automate **Docker image builds** and push to **Docker Hub** on every main-branch commit, achieving **zero-downtime continuous delivery** across all three containerized services.

Bhopal Food Choice — Train-Seat Food Delivery | [React.js](#) · [Node.js](#) · [Express.js](#) · [MongoDB Atlas](#) · [Socket.IO](#) · [JWT](#) · [bcrypt](#) · [Vercel](#) · [Render](#)
[bhopal-food.vercel.app](#) | [25Rohit25/Bhopal-Food](#)

- **What & Uniqueness:** Engineered a production-grade **MERN platform** for **PNR/seat-precise** food ordering on trains, uniquely featuring a real-time **collaborative Group Cart (Socket.IO multi-client sync)** and a **live order-status tracking** system — setting it apart from static restaurant apps through its **railway-domain-specific delivery routing** and glassmorphism-first mobile UI (Tailwind CSS · Framer Motion).
- **Why It Exists:** When **families or friend groups** travel together across **different coaches and berths**, coordinating a single food order becomes **fragmented and chaotic** — each person must order separately, payments split awkwardly, and deliveries arrive at inconsistent times; this platform solves that by letting scattered passengers join one **shared Group Cart** in real time, place a **unified order**, and receive delivery mapped precisely to each member's **PNR and seat number**.
- **How It Solves:** Resolves **multi-user cart collision** via **atomic Socket.IO event broadcasts** ensuring all group members see **consistent cart state** in real time; enforces a **multi-role security model** (Passenger · Admin) with **stateless JWT auth** and **bcrypt-hashed credentials**; deployed on a **decoupled CI/CD pipeline — Vercel (frontend) + Render (backend)** — guaranteeing **zero-downtime** production pushes.

Waste-No-More — Geospatial Food Rescue Platform | [Next.js 14](#) · [TypeScript](#) · [Node.js](#) · [Socket.IO](#) · [MongoDB](#) · [Google Maps API](#) · [JWT](#)
[waste-no-more.vercel.app](#) | [25Rohit25/Waste-No-More](#)

- **What & Uniqueness:** Architected a **distributed event-driven** geospatial food rescue **platform (Next.js 14 · TypeScript · Node.js · Socket.IO · MongoDB · Google Maps API)** uniquely combining **live map-based donation** discovery with **4-role RBAC** (Donor · Volunteer · Receiver · Admin) and **97%+ TypeScript coverage** — distinguishing it from static food-bank directories through **real-time spatial coordination**.
- **Why It Exists:** Millions of kilograms of **surplus food are wasted daily** while volunteers and receivers have **no real-time visibility** into nearby donations; this platform validates the coordination gap by enabling **configurable-radius geospatial queries** that surface active **donations** to the right volunteers the moment they are posted.
- **How It Solves:** Leverages **MongoDB 2dsphere indexed queries** for **sub-millisecond radius-based** donation lookup; eliminates **duplicate-pickup race conditions** through **atomic Socket.IO broadcasts** on claim events that instantly remove **stale entries** from all connected volunteer maps — ensuring **distributed consistency** without additional database writes.

COMPETITIVE PROGRAMMING

Codeforces Expert · CodeChef 5★ · LeetCode Knight · Top 5% · 500+ problems solved — algorithms, graph theory, dynamic programming, system-level problem solving

CERTIFICATIONS

MongoDB Associate Developer · SAP Certified — Generative AI Developer · Oracle Cloud Infrastructure — Architect Associate