

**RUHI CHOPDA** Software Engineer • Full-Stack • Cloud • Microservices • Distributed Systems Bangalore, India

Email: chopdaruhi9@gmail.com Phone: +91 7201096449

LinkedIn: <https://www.linkedin.com/in/ruhi-b-chopda-b70a94308/>

GitHub: [github.com/RuhiChopda](https://github.com/RuhiChopda)

Portfolio:

**SUMMARY** Software Engineer skilled in building scalable cloud-native applications using React, Angular, Node.js, Java,

Spring Boot, Microservices, Docker, Kubernetes, and CI/CD. Strong understanding of distributed systems,

API design, and cloud deployments (GCP/AWS/Azure). Experienced in building production-ready full-stack

applications and cloud-native microservices.

**TECHNICAL SKILLS** Frontend: React.js, Next.js, Angular, TypeScript, TailwindCSS

Backend: Node.js, Express.js, Java, Spring Boot, Microservices, REST APIs, JWT/OAuth2

Databases: MongoDB, PostgreSQL, MySQL, Query Optimization

Cloud & DevOps: GCP (Cloud Run, Compute Engine), AWS (EC2, S3), Azure, Docker, Kubernetes, Terraform, GitHub Actions, Jenkins, CI/CD

Other: System Design, Distributed Systems, Redis Caching, WebSockets (Socket.io)

## PROJECTS

Java + Spring Boot + Microservices – E-commerce Order System

- Built a microservices-based system with Order Service, Payment Service, and Notification Service.

- Used Spring Boot, Java, Eureka Discovery, API Gateway, and Docker.

- Implemented asynchronous communication with RabbitMQ and fault tolerance with Resilience4j.

- Deployed on GCP Cloud Run using containerized services.

Angular Inventory Dashboard – Full Frontend SPA

- Developed a dashboard with Angular, RxJS, Material UI, and API integration.

- Implemented module-based routing, state management, charts, and CRUD UI.

- Improved load performance using lazy loading and optimized components.

GCP Deployment Project – Cloud-Native Image Recognition API

- Containerized an ML model and deployed on GCP Cloud Run.

- Configured CI/CD with GitHub Actions and automated image builds.

- Integrated Cloud Storage for datasets and used Cloud Logging for monitoring.

Encrypted Health Chat App (MERN + Microservices + Steganography)

- Secure chat system embedding encrypted messages into images.

- Used distributed microservice architecture and optimized DB latency by 40%.

### Real-Time Admin Dashboard (React + WebSockets + Cloud Deployment)

- Implemented real-time analytics using Socket.io and role-based access.
- Deployed using Docker, CI/CD, and caching to improve API speed by 35%.

### AI Plant Disease Detector (Python + CNN + Flask)

- Built ML inference backend with <350ms latency and deployed as containerized API.

### Steganography-based Secure Communication System (MERN)

- Created a secure multimedia communication pipeline with encrypted channels.

**EDUCATION** B.Tech in Computer Engineering (Software Engineering) Jain University, 2022–2026

### **SYSTEM DESIGN & CORE CS** - Microservices, API gateways, load balancing

- Caching, rate limiting, horizontal scaling
- Kubernetes orchestration
- Design patterns (Singleton, Factory, Observer)
- DSA: graphs, DP, trees, greedy