

SAMARTH CHANDRA

+91-9608621700
samarthchandra.in

 samarth.chandra07@gmail.com

 github.com/samarth-devx

 linkedin.com/in/samarth-chandra

Education

Noida Institute of Engineering & Technology

Bachelor of Technology - CSE; CGPA: 8.92

Courses: Operating Systems, Data Structures & Algorithms, DBMS, Cloud Computing, Computer Networks

Greater Noida, India

2024 - Present

Skills

- **Languages:** Java, C++, JavaScript, TypeScript
- **Backend & Frameworks:** Spring Boot, NodeJS, WebSockets, REST APIs
- **Cloud, DevOps & Databases:** AWS, Docker, PostgreSQL, Redis, Kafka, Git, CI/CD
- **Software Engineering:** Object-Oriented Programming (OOP), System Design, Agile

Projects

- **AutoShip - Automated build & deployment pipeline**   - Nodejs, Docker, AWS, PostgreSQL, Kafka
 - Enabled 1-click cloud deployments for applications by building a platform that performs repo cloning, builds, & hosting.
 - Cut API Server load exponentially by offloading all build execution to isolated ECS Fargate containers.
 - Scaled deployments horizontally and reliably by parallelizing isolated builds across ephemeral Docker containers.
 - Persisted build artifacts to S3, and abstracted all cloud infra complexity from users via proxy-routing project subdomains.
 - Delivered real-time build visibility logs to users via Kafka for durable messaging and Socket.IO for live updates.
 - Optimized high-volume log ingestion via ClickHouse, while managing project metadata in PostgreSQL via Prisma.
- **BitFlow - Video transcoding pipeline**   - Nodejs, FFmpeg, Docker, AWS SQS, S3
 - Automated scalable video transcoding by building a pipeline that converts raw uploads into 3 output variants (360p, 480p, and 720p) using AWS ECS Fargate and FFmpeg.
 - Decoupled uploads from processing by using S3 presigned URLs for direct client uploads.
 - Queued transcoding jobs automatically via S3 event notifications triggering SQS messages.
 - Achieved scalable, burst-tolerant video processing by using Amazon SQS long polling to provision on-demand ECS Fargate containers, each executing multiple FFmpeg transcoding jobs concurrently.
- **MeshPay - Offline P2P payment system**  - Java, Spring Boot, Spring Data JPA, Hybrid encryption (RSA/AES)
 - Built an offline-first P2P payment system enabling secure digital payments without internet connectivity.
 - Extended payment reach beyond Bluetooth range by routing encrypted transactions across intermediate mesh nodes.
 - Ensured end-to-end secured transaction using hybrid RSA-2048 and AES-256-GCM encryption
 - Prevented intermediaries from accessing or modifying data, via layered payload encryption.
 - Guaranteed exactly-once settlement through ciphertext-hash idempotency across redundant mesh deliveries.

Certifications and Awards

- **Google Cloud AI Infrastructure**  : AI Hypercomputer architecture, GPU/TPU acceleration, and infrastructure optimization strategies
- **Google Agile**  : Agile principles, Scrum framework, and sprint-based iterative project execution.

Engineering Articles

- How Weverse works across the insane traffic of K-pop fans. 
- Inside the engineering of UPI: India's biggest payment infrastructure. 
- Distributed systems from First Principles. 