

AASHISH KUMAR

+918084662242 aashishkr.iitkgp@gmail.com linkedin.com/in/kr-aashish github.com/kr-aashish

Education

Indian Institute of Technology Kharagpur

Jun 2018 – May 2023

Dual Degree (B.Tech & M.Tech), CGPA: 8.52/10

Kharagpur

Technical Skills

Programming Languages: C++, Java, Golang, Python, JavaScript, TypeScript

Markup/Query Languages: HTML5, CSS3, T-SQL, pgSQL

Frameworks & ORMs: Spring Boot, Gin, Node.js, Flask, jOOQ, Mongoose, Prisma, Gorm

Databases: Redis, MongoDB, DynamoDB, Elasticsearch, SQLite, SQL Server, PostgreSQL, pgvector

Cloud: S3, EC2, RDS, CloudWatch, Lambda, SNS, SQS, SES, ECS, Docker, IIS, Nginx, GitHub Actions, Azure CI/CD

Experience

FlashMonk

Dec 2024 - Present

Software Development Engineer

Bangalore

- Redesigned the IMAP service, improving wait times by 16% through event-based mail fetch execution, auto-retry mechanisms, and configurable domains.
- Improved the email compose experience flow, reducing errors by 30%, and migrated mail view APIs from a database to Elasticsearch, which reduced mailbox latency by 25%.
- Migrated to an Affiliate Campaign and reward system, including campaign execution, audience segmentation, and click-through analytics, reducing customer engagement costs by 80%.
- Developed a Collections Data Ingestion Service, Collections Dimension Service, and a Collections Central Go Orchestrator, utilizing a Vector DB, ELSER, ES Mappings, and Product/Brand Semantic Search.
- Migrated 40 microservices from CloudWatch to Last9 with OTEL, achieving a 30% reduction in costs, right-sized ES/RDS clusters, dropped a 300M index, and decommissioned 2 redundant RDS instances and 1 EFS.
- Re-engineered email info nested partitions, deleted 3 billion promotional email records from S3 and the database, and migrated Mime RDS to S3, resulting in a 30% reduction in RDS costs.

RootQuotient

Jun 2023 - Nov 2024

Software Development Engineer

Chennai

- Created Data Service APIs to migrate a Cancer Registry System, from Oracle PL/SQL Stored Procedures & Triggers to TS, leveraging SQL Server & Prisma
- Migrated .NET Web Services by creating a streaming RPC client & SOAP API server with WSDL, in Node.js, which consumes DLLs via wrappers using FFI
- Scheduled cron job for consuming HL7 documents, with transactional locks, and a configurable job interval, scaled to Node.js cluster threads through IPC
- Wrote unit & integration test suites, with a statement & branch coverage of 85%, optimized on-premise infra requirements with Grafana's K6 perf tests
- Configured build pipeline with coverage reporter & static code analyzer, and release pipeline with Azure agents, deploying in Windows servers with IIS

Amazon

May 2022 – Jul 2022

Software Development Engineer Intern

Bangalore

- Designed a UUID for the CGS Dashboard versioning and improved cadence support from weekly to every second, in the Consolidation EMR's ETL job
- Revamped the Audit Log DynamoDB to store all the metadata for the CGS Dashboard, tracking the published versions, reflecting in the CGS Version UI
- CGS UI: Enhanced with filters, publish time, and the EMR status display
- Reengineered the CSV View & Diff modules to render the dashboard directly from S3, eliminating the cost of maintaining the Quarterly Competitor DDB
- Implemented a CSV diff algorithm to compare the dashboard versions by the given primary index, providing a compact list of changes to the stakeholders

Projects

Jira 2.0 | *Autonomous Project Management Suite* | [View System Design](#)

Apr 2024 – Jun 2024

- Crafted a Lambda which triggers upon the project's FRD upload into S3, and creates user stories and tasks by leveraging Llama3 running in a private EC2
- Facilitated Event Bridge to consume Slack messages periodically, then used the LLM through Flask APIs to compile task's status and assignment updates
- Employed SES to dispatch task update approvals to stakeholders, thereupon invoking issue tracker's APIs to reflect corresponding changes in the backlog

DeployX | *UI Deployment Automation* | [Design Reference](#)

Jan 2024 – Mar 2024

- Built an API server with a private RDS, which queues build jobs, and streams logs to Kafka for ingesting into ClickHouse, accessible as a polling endpoint
- Provisioned an auto-scaled ECS cluster, which runs an ECR image in Fargate, to process the build job for the given codebase, and dump the artifacts to S3
- Developed a reverse proxy for the deployed UI builds, to stream static assets from S3, mapping the projects's subdomain to the S3 key with an RDS query

Aranpani | *Fundraiser Management System* | [OpenAPI Docs](#)

Nov 2023 – Dec 2023

- Modeled project, donor & payment endpoints, with JWT auth, serializer, Joi validation middleware layers, adopted S3 pre-signed URLs for media uploads
- Penned UTs for models & controllers and acceptance test suites for the API endpoints, maintaining a 90% above code coverage, with Jest and SuperTest
- Ensured application's reliability with MongoDB Replica Sets, in PM2 cluster mode with APM, leveraging Nginx as a web server and Route53 name server

Coursework & Certifications

- | | | |
|---------------------------------|----------------------------------|--|
| • Programming & Data Structures | • Data Analytics | • AlgoExpert algoexpert.io |
| • Algorithms 1 | • Machine Learning Foundations | • FrontendExpert algoexpert.io |
| • Applied Algorithms and DS | • Deep Learning Foundations | • SystemsExpert algoexpert.io |
| • Probability and Statistics | • Winter School of AI & Robotics | • InfraExpert algoexpert.io |