Anudeep Allamsetty

Portfolio| LinkedIn| allamsettyanudeep.dev@gmail.com |Phone: (913)-303-1252| GitHub

PROFESSIONAL SUMMARY

- **Java Full Stack Engineer** with **5+** years of experience designing and deploying enterprise-scale applications across **insurance**, **banking**, **pharma**, and **manufacturing domains**. Skilled in **Java 17, Spring Boot, Angular 17, React, and cloud-native architectures on AWS and Azure**, specializing in **microservices**, **CI/CD automation**, and full lifecycle **SDLC delivery** using **Agile** and **DevOps practices**.
- Architected and **migrated legacy claim** and inventory systems into **Spring Boot microservices** hosted on **AWS EC2**, **S3**, and **Lambda**, implementing domain-driven boundaries, **async workflows** via **Kafka Streams** and **RabbitMQ**, and containerized rollouts using **Docker**, **Kubernetes**, and **Helm**. Improved scalability by **60%**, reduced deployment time by **75%**, and eliminated configuration drift across environments through Terraform automation.
- Developed and optimized **RESTful** APIs using **Java 17**, **Spring** MVC, and **Spring Security** (**JWT, OAuth2, Okta**) with **DTO mapping**, global exception handling, and **API** documentation via **Swagger** / **OpenAPI**. Reduced integration defects by 30%, enforced **RBAC** across distributed services, and strengthened audit compliance with **Spring Batch**, **JMS**, and custom validation interceptors.
- Engineered responsive front-end modules using **Angular 17** (**NgRx**, **AG Grid**) and **React** (**Redux Toolkit**, **React Hook Form**), implementing **lazy loading**, **server-side pagination**, and **virtual scrolling** for **1M+ record datasets**. Deployed optimized builds via **Node.js / NPM pipelines**, integrated **Cypress** / **Playwright regression** suites, and improved **UI performance** by **40%** while reducing frontend bugs by **25%**.
- Automated delivery, monitoring, and observability pipelines using **Jenkins**, **Azure DevOps**, **GitHub Actions**, **Terraform**, and **SonarQube**, **maintaining** approx. **90%** code coverage with **JUnit 5**, **Mockito**, and **REST** Assured. Integrated **Open Telemetry**, **Prometheus**, and **Grafana** for distributed tracing and alerting, cutting mean-time-to-resolution by 50%, and optimized data caching via **Redis** and **PL/SQL** tuning for consistent sub-second API response times.

PROFESSIONAL EXPERIENCE

Client: **Zurich Insurance Group, Overland Park, Kansas, USA**

Role: **Software Engineer** **Jan 2024 – Present**

- Architected and **Transitioned legacy** claim modules into **Spring Boot 3** microservices on **AWS** (**EC2, S3, RDS**) using domain-driven design and **Spring Cloud Config**. Improved scalability by **60 percent** and **reduced release time** from **two hours to forty-five minutes.**
- Crafted **RESTful APIs** in **Java 17** with **DTO mapping**, layered **exception handling**, and **Swagger** documentation. Reduced integration fault by 30 percent and accelerated **cross-team onboarding**.
- Developed **Angular 17** dashboards with **NgRx** store, **lazy-loaded modules**, and automated **Node.js** build pipelines. Reduced **API over fetching** and Boosted **UI responsiveness** by **40 percent** during peak usage.
- Integrated **AG Grid** with **server-side pagination**, infinite scroll, and virtual row models to render more than one million policyholder records efficiently, maintaining **sub-800 ms** response times.
- Optimized **backend** throughput by implementing **Redis caching** for frequently accessed **claim metadata**, **lowering latency** from **1.2 seconds to 0.45 seconds** and improving concurrent handling by 35 percent.
- Secured distributed **microservices** with **Spring Security** (**JWT, OAuth2**) and **Okta SSO**, enforcing granular **RBAC** policies and preventing unauthorized access incidents in production.
- Orchestrated **AWS** provisioning with **Terraform** for **IAM**, **EC2**, and **VPC** resources, ensuring reproducible environments and eliminating manual configuration errors across development and test pipelines.
- Containerized and deployed **Spring Boot** services with **Docker** and **Kubernetes** using **Helm charts** configured for rolling updates, **health probes**, and **auto-scaling**. Augmented uptime and deployment reliability by 40 percent.
- Established observability by integrating **AWS CloudWatch** and **AppDynamics dashboards** for **JVM metrics** and request tracing. Reduced **mean time** to resolution **by 50 percent** through **Preventive alerts** and profiling.
- Implemented event-driven workflows using AWS Lambda and SQS for asynchronous claim notifications and background validation. Decreased manual reprocessing effort by 45 percent and Amplified message throughput.
- Led Agile sprint ceremonies in Jira, facilitated retrospectives, and mentored engineers on SOLID principles and peer reviews. Boosted team velocity by 20 percent and maintained SonarQube coverage above 90 percent.

Client: **Seaboard Corporation, Merriam, Kansas, USA**

Role: **Java Full Stack Developer** **Sep 2023 – Dec 2023**

- Crafted a real-time commodity pricing dashboard using **React.js** with **Redux** Toolkit and **Thunk**, streaming **Kafka updates** and **backend API** data to maintain consistent state **synchronization**. Enabled instant price refresh and Boosted data visibility across global trading desks.
- Constructed **Spring Boot microservices** backed by **PostgreSQL** using **ModelMapper** for **DTO transformation**, structured exception handling, and **service-layer** validation. Streamlined request processing, **improving API response times** by **25 percent** under concurrent load.
- Integrated **RabbitMQ queues** between order and **inventory microservices** to **decouple processing logic** and ensure reliable message delivery. Amplified throughput by 35 percent and prevented message loss under heavy transaction volume.
- Migrated legacy **Angular modules** to a **modular React component** architecture with **TypeScript** and **Hooks**, **reducing UI defect rates by 25 percent** and improving maintainability for future enhancements.
- Executed modular administrative forms using **React Hook Form** with custom validation schemas and reusable components, improving **data-entry accuracy**, and reducing **client-side validation** errors by 30 percent.
- Streamlined **SQL stored procedures** and joins for shipment **aggregation workflows**, lowering **dashboard** load time from seven minutes to two minutes during peak operating hours.
- Delivered the **React front end** to **Azure Blob Storage** with **CDN caching**, achieving a 45 percent reduction in global load latency and simplifying multi-region release management.
- Configured **Azure DevOps** pipelines with gated approvals, rollback triggers, and artifact versioning to ensure deployment safety during continuous delivery cycles.
- Integrated legacy partner systems via **SOAP services** using **JAXB bindings**, maintaining backward compatibility with external **vendor APIs** while **migrating** to **RESTful endpoints**.
- Formulated **Cypress regression** test suites covering **core UI flows**, integrated with **CI pipelines** to detect visual and functional issues early. Reduced **manual QA** time by **40 percent** and increased release confidence across sprints.

Client: **Cognizant (Key Bank), India**

Role: **Java Full Stack Developer** **Aug 2021 – May 2022**

- Modernized **legacy Struts** modules into **Spring Boot REST APIs** using **DAO** and **DTO** patterns with layered architecture, improving maintainability and **unit-test coverage** while reducing **technical debt by 40 percent**.
- Developed customer onboarding forms with **React.js** and **Context API**, implementing multi-step navigation and shared state validation. Upgraded data accuracy and reduced form submission errors by 30 percent across customer workflows.
- Established **Spring Security** with **JWT** and **CSRF** protection to secure authentication and session flows for public-facing banking modules. Eliminated recurring **unauthorized access** issues during **UAT testing cycles**.
- Architected and optimized **SQL stored** procedures for **audit logging** and **customer verification**, streamlining backend processing and **reducing query execution** time by **25 percent** during form submissions.
- Produced **unit** and **integration test suites** using **JUnit** 5 and **Mockito**, mocking service and repository layers to achieve over 90 percent code coverage and decrease regression Error rate by 60 percent.
- Rolled out **containerized microservices** to **internal Kubernetes clusters** through **Helm charts**, supporting **blue-green testing strategies** and improving environment parity across **QA** and **staging**.
- Integrated **DynamoDB** using **Spring Data** for **Real-time customer metadata storage**, adopting **schema-flexible** modeling that accelerated feature delivery and simplified version management.
- Collaborated with **cross-functional team**s in **daily Agile standups**, coordinated Bug density triage with **QA**, and reviewed pull requests with **senior developers** to ensure stable UAT releases and timely sprint closures.
- Maintained legacy **JSP** and **jQuery** user interfaces while progressively **migrating** components to **React**, reducing maintenance **overhead**, and improving **UI consistency** across platforms.

Client: **Solara Active Pharma Sciences Ltd, India**

Role: **Java Developer** **Mar 2019 – Jul 2021**

- Formulated and produced inventory restocking and compliance tracking pipelines using **Spring Batch** with custom schedulers, automating **manual QA tasks**, and reducing operational **workload by 70 percent** across production teams.
- **Programmed** full-stack clinical trial management modules using **Spring Boot** and **React.js**, implementing audit trail logging, **RBAC**, and validation layers to maintain compliance with industry regulatory standards.
- Integrated **MongoDB** with **Spring Data** for managing **unstructured clinical** notes and configured **TTL indexes** for Systematized archival of outdated records, ensuring data integrity and reducing storage usage by 15 percent.

- Refactored legacy approval workflows from **JSP** and **Struts** into **REST-driven React** interfaces, improving **UI load times** by **35 percent** and reducing **post-deployment** post release issues rates by **25 percent**.
- Created and Refined **PL/SQL** Managed procedures for **audit exports** and regulatory reporting, **cutting query** execution time by over **60 percent** and improving overall **data accuracy**.
- Utilized **Docker Compose** to simulate **microservice environments** locally, **reducing setup time** for new developers by **50 percent** and improving **cross-functional** collaboration during **testing cycles**.
- Produced **Groovy** and **Bash automation scripts** for build **validation**, **log cleanup**, and health checks in **Jenkins CI pipelines**, reducing **manual QA** validation time by **80 percent** and increasing release consistency.
- Refactored scheduled batch jobs to **Spring Scheduler** using **JDBC templates**, enhancing **error tracking**, retry handling, and overall job stability in production environments.
- Engineered **multi-threaded backend workflows** using **JMS** with **retry logic** and **dead-letter queues**, achieving fault-tolerant batch execution with high throughput and zero message loss under load.
- Participated in **Agile sprints** and **retrospectives**, Authored **UML** design documentation for new modules, and ensured complete handoff packages for smooth transitions between teams.

EDUCATION

- **University of Central Missouri,** *Missouri, United States*
  Master of Science in Computer Science

CERTIFICATIONS & ACHIEVEMENTS

- AWS Certified Solutions Architect – Associate - LINK
- Microsoft Certified: Azure Administrator Associate - LINK
- Oracle Certified Professional: Java SE 11 Developer - LINK

TECHNICAL SKILLS

- **Languages:** Java (8–17), TypeScript, JavaScript, SQL, Bash, Python, XML, JSON
- **Frontend Technologies:** Angular (13–17), React.js (Hooks, Redux Toolkit, Context API, React Hook Form), HTML5, CSS3, SCSS, Bootstrap, Tailwind CSS, AG Grid, JSP, Struts
- **Backend & Frameworks:** Spring Boot, Spring MVC, Spring Security (JWT, OAuth2, Okta), Spring Batch, Spring Scheduler, Spring Data JPA, Hibernate, REST APIs, Microservices, Model Mapper, DAO/DTO Patterns
- **Cloud Platforms:** AWS (Lambda, SQS, API Gateway, EC2, S3, IAM, CloudWatch), Azure (Blob Storage, Azure DevOps Pipelines), Kubernetes
- **DevOps & Build Tools:** Docker, Docker Compose, Kubernetes, Helm, Terraform, Jenkins, Azure DevOps, GitHub Actions, Maven, Gradle
- **Databases:** PostgreSQL, MySQL, Oracle (PL/SQL), MongoDB, DynamoDB, Redis
- **Messaging & Streaming:** Apache Kafka (Kafka Streams), RabbitMQ, JMS
- **Testing & QA:** JUnit 5, Mockito, Cypress, Playwright, REST Assured, Postman
- **Monitoring & Logging:** AppDynamics, ELK Stack (Elasticsearch, Logstash, Kibana), AWS CloudWatch, Azure Monitor
- **Documentation & Modeling:** Swagger/OpenAPI, UML (Use Case, Class, Sequence Diagrams)
- **Version Control & CI/CD:** Git, GitHub, GitLab, Bitbucket
- **Processes & Methodologies:** Agile (SCRUM), Sprint Planning, PR Reviews, TDD, BDD, Code Reviews
- **AI & GenAI (Academic and Practical Exposure):** Prompt Engineering, OpenAI API, LangChain, Hugging Face (basic), GPT-4, Retrieval-Augmented Generation (RAG), AutoGen

Projects:

- **Job Board Application**
  Designed and Created job tracking platform with role-based authentication. Recruiters can post/delete jobs; candidates can sign up, apply, and track applications. Integrated secure JWT auth, REST APIs, and responsive UI using Tailwind.
- **Expense Tracker App**
  Built a personal expense management app with custom UI. Users can track income/expenses, categorize transactions, and visualize spending. Backend includes robust RESTful API with Spring Boot and MySQL data persistence.