DAKSH PRUTHI

+1 (201) 284-8928 | dakshpruthicareers@gmail.com | Jersey City, NJ, USA | linkedin.com/in/dakshpruthi/ | github.com/dakshpruthi099 | www.dakshpruthi.com/

PROFESSIONAL SUMMARY

Full-Stack Java Software Developer with ~3 years of experience focusing on Spring Boot, Microservices, and REST APIs. Adept at architecting cloud native applications using AWS, Kafka, Redis, Docker, and Kubernetes. Experienced in database optimization, CI/CD pipelines, and bridging technical expertise with business needs to deliver user-centric solutions.

SKILLS

- **Programming Languages:** Java, JavaScript, Python, SQL, Bash
- **Frameworks:** Spring Boot, Hibernate, Node.js, React.js, Next.js
- **Databases / Messaging:** MySQL, PostgreSQL, MongoDB, Pinecone, PGVector, Redis, Kafka
- **Cloud / DevOps:** AWS (EC2, S3, Lambda, API Gateway), Docker, Kubernetes, Terraform, CI/CD (Jenkins, GitHub Actions)
- **Technologies / Tools:** OpenAI API, OAuth, REST APIs

PROFESSIONAL EXPERIENCE

Sprowt
Software Development Engineer *September 2024 - Present*

- Architected a resilient Java Spring Boot microservices ecosystem for a workforce management platform, leveraging Kafka for real-time event streaming to ensure high availability and seamless scalability, achieving 99.99% uptime.
- Engineered a multi-threaded recommendation engine to optimize employee scheduling, leveraging Java 17, Spring Boot, and MongoDB, reducing response times by 60% and improving real-time user interactions.
- Implemented IaC strategies using AWS CDK, automating deployment pipeline to AWS ECS, Lambda, and RDS with zero downtime.
- Integrated Redis caching to reduce query latency by 40%, speeding up analytics and decision-making for managers.

Bespoke Digital Media
Software Developer *April 2021 - July 2022*

- Led a team of 2 developers to configure a Java Spring Boot-based CMS for media agencies, serving 50K+ concurrent users with Redis caching and database optimizations, achieving 30ms API response times and 99.9% uptime.
- Implemented authentication with RBAC and OAuth2.0 in Spring Security, ensuring enterprise-level security standards.
- Applied ACID-compliant transaction handling in PostgreSQL, ensuring data integrity for reliable content publishing.
- Engineered a real-time WebSocket analytics dashboard using Spring Boot and Redis Pub/Sub, processing 100K+ events per minute with minimal latency for client campaign tracking.

FIS Global
Senior Analyst *July 2019 - March 2021*

- Designed and implemented a payment gateway with Java Spring Boot, supporting $70M+ daily transactions with Kafka-based event processing for fault-tolerant, low-latency services.
- Led migration to microservices architecture, addressing scalability issues and boosting system performance by 70%.
- Reduced transaction latency by 35% by optimizing Redis caching and SQL queries, addressing peak-hour bottlenecks.
- Achieved 90% test coverage for core modules using JUnit, Mockito, and REST-Assured for comprehensive API testing.
- Spearheaded the use of Jenkins and Docker for automating CI/CD pipelines, improving release cycle efficiency by 20%.

EDUCATION

Stevens Institute of Technology **September 2022 - May 2024**
Master's, Computer Science *GPA: 3.63*

- Courses: Database Management, Java, Operating Systems, Data Structures and Algorithms, Cloud Computing

Bharati Vidyapeeth University **July 2015 - June 2019**
Bachelor's, Computer Engineering *GPA: 7.72*

PROJECTS

Event-Driven Payment Gateway

- Built a Java Spring Boot payment gateway with Kafka for real-time event processing, reducing failures by 30%
- Secured payment transactions by integrating OAuth2.0 and JWT for token-based authentication.
- Resolved database bottlenecks implementing indexing and sharding, reducing load times by 40% for high volume transactions.

E-commerce Microservices Platform

- Designed a scalable microservices platform using Java, Spring Boot, Docker, and Kubernetes, cutting cart abandonment by 25% with Redis and Kafka-driven recommendations.
- Resolved peak period performance issues by auto-scaling Kubernetes pods, ensuring <1s response for 10K+ concurrent users.
- Integrated real-time payment tracking and fraud detection algorithms using Spring Security and JWT, preventing fraud.