

ABHINAV BHARGAV

Email: abhinavbhargav533@gmail.com | Phone: +91 7992378150

Java Backend Developer

SUMMARY

Backend Developer with 3.5 years of professional experience in designing, developing, and deploying scalable backend applications using Java, Spring Boot, and Microservices Architecture. Strong expertise in building RESTful APIs, implementing JWT-based authentication and Role-Based Access Control, and developing Event-Driven systems using Apache Kafka. Hands-on experience with AWS (EC2, S3, IAM) and Docker for cloud-native deployments. Proficient in Agile (Scrum) methodology, collaborating with cross-functional teams using JIRA, and Integrated third-party services for payments, email, and SMS notifications to deliver secure, high-performance enterprise applications.

Technical Skills

Programming & Frameworks

Java 8, Spring Boot, Spring MVC, Spring Security, Spring Cloud, Hibernate (JPA), Feign Client, RESTful APIs, Microservices Architecture , Spring Batch

Database & Caching

MySQL, Redis, ER Diagrams

Cloud & DevOps

AWS (EC2, S3, IAM), GitHub Actions

Messaging & Communication

Apache Kafka, Feign Client, Eureka Server, API Gateway

Authentication & Security

JWT, Spring Security, Role-Based Access Control (RBAC)

Tools & Platforms

JIRA, Postman, Swagger, IntelliJ IDEA, Spring Tool Suite (STS), Maven, Gradle, Git, GitHub

Monitoring & Logging

SLF4J, Zipkin, Spring Boot Admin, JUnit

Development Process

Agile environment – Sprint Planning, Daily Stand-ups, Reviews, Retrospectives

WORK EXPERIENCE

Java Developer , EBOT TECHNOSOFT

3.5 years

EDUCATION

Bachelor of Technology (B.Tech) – Computer Science Engineering
ITER, SOA University

WORK EXPERIENCE

Java Backend Developer – EBOT TECHNO SOFTWARE

Client: lastminute.com

Tech Stack: Java, Spring Boot, Spring Security, Hibernate (JPA), Apache Kafka, Feign Client, AWS (EC2, S3), Docker, Redis, MySQL, Stripe SDK, SendGrid, Twilio, GitHub, Maven, SLF4J.

• Modules & Responsibilities

1. User Service

- Implemented secure login, registration, password management.
- RBAC(Admin/Customer) using Spring Security + token-based authentication.
- Persistent session handling with token validation.
- Used SLF4J for structured logging and debugging.

2. Hotel Service

- CRUD operations for hotel listings, metadata (images, rules, amenities).
- Managed image uploads via AWS S3 and exposed presigned URLs.
- Used Hibernate/JPA for relational mapping and validations.

3. Room Inventory Service

- Handled availability, room types, and dynamic pricing logic.
- Implemented date-based inventory tracking with optimized queries.
- Integrated Redis for caching frequently accessed inventory data.

4. Booking Service

- Developed a booking engine with validation, availability check, and confirmation number generation.
- Apache Kafka used to broadcast booking events (to Payment, Notification services).
- Synchronized inventory update and cancellation workflows.

5. Payment Service

- Integrated with Stripe SDK (sandbox & production) for secure payments.

6. Search & Filter Service

- Implemented full-text search with Elasticsearch by hotel name, location, amenities.
- Applied multi-criteria filters: star rating, price range, facilities, room type.

7. Review & Rating Service

- Enabled users to add/edit/delete hotel reviews and ratings.
- Aggregated scores used in hotel ranking algorithm.

8. Notification Service

- Real-time communication via Apache Kafka, triggered SMS/email using Twilio/SendGrid.

9. Infrastructure & Integration

- Implemented API Gateway for routing, security, and rate limiting.
- Eureka Server for service discovery and registration of microservices.
- Zipkin integration for distributed tracing and performance monitoring.
- Spring Boot Admin for real-time monitoring of service health.
- Terraform used for provisioning AWS infrastructure (EC2, S3, IAM roles).
- Dockerized all services; deployed via Kubernetes (EKS) for production scaling.
- Created ER Diagrams and UML class diagrams for design reviews.
- Used Feign Client for inter-service REST communication with fallback logic.