

Ayshi Shannidhya Panda

Java Backend Developer • Spring Boot • Microservices Engineer

✉ asp45624@gmail.com

☎ +91 8658472229

🌐 ayshishannidhya.online

🐙 github.com/ayshishannidhya

🌐 linkedin.com/in/ayshishannidhya

Professional Summary

I am a Java Backend Developer focused on building scalable, secure, and high-performance microservices. I work primarily with Spring Boot, Spring Security, and Spring Data JPA, and I have hands on experience designing event-driven systems using gRPC and RabbitMQ. I follow clean architecture and domain-driven design principles to write maintainable, production-grade code. I enjoy solving complex backend problems around distributed systems, inter-service communication, and real-time data flow.

Technical Skills

- **Languages:** Java 21, SQL
- **Frameworks:** Spring Boot, Spring WebFlux, Spring Security, Spring Data JPA, Hibernate, Spring Cloud
- **Databases:** PostgreSQL, MySQL, Firebase
- **DevOps & Tools:** Docker, Git, Maven, Gradle, JUnit, Mockito, Postman, Linux
- **Messaging & Communication:** RabbitMQ, gRPC, WebSocket (STOMP)
- **Monitoring:** Spring Boot Actuator, Micrometer, Spring Boot Admin
- **Concepts:** Microservices, Clean Architecture, Domain-Driven Design, Event-Driven Architecture, Reactive Systems, Distributed Systems, RESTful API Design

Projects

Patient Management System — Microservices Healthcare Platform — [GitHub](#)

Spring Boot 3.4, gRPC, Docker, Gradle

- Architected 3 microservices (patient-service, billing-service, analytics-service) across 8 packages with Gradle multi-project build and Dockerized deployment
- Integrated gRPC with Protobuf for inter-service billing communication, eliminating REST overhead for internal calls
- Developed 4 RESTful API endpoints with Spring Data JPA, Bean Validation, DTO mapping layer, and auto-generated API docs using SpringDoc OpenAPI (Swagger UI)

Neptune Bank — Microservices Banking System — [GitHub](#)

Spring Boot 3.5, JWT, RabbitMQ, Eureka, PostgreSQL

- Designed 5 independently deployable microservices (auth, user, account, transaction, OTP) with Spring Cloud Netflix Eureka for service discovery and load balancing
- Implemented JWT-based authentication (jjwt) with Spring Security across all 5 services; integrated Spring Boot Admin for centralized health monitoring
- Connected account-service to RabbitMQ via Spring Cloud Stream Binder for asynchronous event-driven messaging, decoupling inter-service dependencies

Messenger — Real-Time Chat & Communication Backend — [GitHub](#) Spring Boot 4.0, WebSocket, STOMP, Spring Security, PostgreSQL

- Built real-time messaging backend using WebSocket with STOMP protocol and SockJS fallback; implemented presence tracking broadcasting ONLINE/OFFLINE events to /topic/presence
- Developed user onboarding flow with 3 REST endpoints and OTP verification via dual-channel delivery (SMS through TextBee API with @Async and email via Spring Mail SMTP)
- Secured all endpoints with Spring Security and WebSocket auth interceptor; persisted user and OTP data with expiry tracking using Spring Data JPA with PostgreSQL

Education

Bachelor of Technology (B.Tech) in Computer Science & Engineering

Silicon Institute of Technology, Sambalpur

2023 – Present (6th Semester)

CGPA: 8.50 / 10.0

Certifications

- **Programming in Java** — IIT Kharagpur (NPTEL) **Elite, 94%**
- **Database Management System** — IIT Kharagpur (NPTEL)
- **Learn Java Programming – Beginner to Master** — Abdul Bari (Udemy)
- **Java Spring Framework, Spring Boot, Spring AI** — Telusko (Udemy)

[Verify](#)

[Verify](#)

[Verify](#)

[Verify](#)