

# TASNIMUL MOHAMMAD FAHIM

+880-193-490-8343 | [faahim06@gmail.com](mailto:faahim06@gmail.com) | [linkedin.com/in/insertfahim](https://linkedin.com/in/insertfahim) | [github.com/insertfahim](https://github.com/insertfahim) | [fahimbuilds.me](https://fahimbuilds.me)

## Technical Skills

**Backend Development & Languages:** Java, Kotlin, Python, JavaScript, Node.js, RESTful API Design, Microservices  
**Frameworks & Ecosystems:** Spring Boot, Spring Framework, FastAPI, Express.js, React.js  
**Databases & SQL:** PostgreSQL, MySQL, MongoDB, SQL Optimization, Database Design, Supabase  
**Event-Driven Architecture:** Kafka, RabbitMQ, SQS, Event Streaming, Message Queues  
**DevOps & Tools:** Docker, CI/CD Pipelines, GitHub Actions, Git, Linux, AWS, Kubernetes basics  
**Development Practices:** Agile/Scrum, Unit Testing, Integration Testing, Code Review, Debugging, TDD

## Education

### BRAC University

*Bachelor of Science in Computer Science and Engineering*

Dhaka, Bangladesh

2025

## Professional Experience

### Software Engineer

Sep 2025 - Present

*The Flex & base360.ai (Sister Companies), United Kingdom*

*Remote*

- Developed backend APIs using Node.js and PostgreSQL serving 1,000+ daily users and processing 5,000+ document transactions monthly, implementing Supabase RLS policies and signed URL generation achieving sub-500ms response times for secure file operations
- Optimized database query performance through indexing strategies and SQL query optimization, reducing document retrieval latency by 60% while maintaining 99.5% system uptime and collaborating with QA engineers to resolve production issues
- Collaborated cross-functionally with product managers and frontend engineers implementing RESTful APIs for contract management features, participating actively in sprint planning, code reviews, and agile ceremonies delivering 15+ features on schedule

### Full Stack Developer

Jun 2023 - Feb 2024

*Doin Tech*

*Remote*

- Built full-stack applications in JavaScript stack (React.js, Node.js, PostgreSQL) handling 500+ concurrent users, designing normalized database schemas with proper indexing and implementing JWT-based authentication for secure user sessions
- Implemented RESTful API endpoints for content management systems, contributing to microservices architecture that improved deployment frequency by 35% and reduced average response time by 45% through caching and optimization
- Accelerated feature delivery through agile sprint methodology, adapting implementations based on peer code reviews and QA feedback while maintaining unit test coverage above 70% and documenting technical specifications

## Key Projects

### IntelliMail - Core Infrastructure Email Service — Python, FastAPI, PostgreSQL, Docker, Redis

[GitHub](#)

- Architected production-grade RESTful API service using Python FastAPI with PostgreSQL backend processing 10,000+ email operations monthly with sub-200ms response times, implementing asynchronous task queues with Celery and Redis
- Implemented optimized database schema utilizing B-tree indexing and foreign key constraints for relational integrity, achieving 80% query performance improvement through JOIN optimization and query plan analysis
- Deployed using Docker containerization and CI/CD automation: GitHub Actions workflows for testing and deployment, achieving 99.9% uptime with 85% automated test coverage and comprehensive API documentation

### ScaleShort - Distributed URL Infrastructure Platform — Java, Spring Boot, PostgreSQL, Redis, Docker

[GitHub](#)

- Developed microservices-based URL shortening system using Spring Boot and PostgreSQL handling 5,000+ daily redirects with sub-50ms latency through Redis caching layer and HikariCP connection pooling for database optimization
- Architected scalable backend infrastructure for high-availability requirements, reducing database load by 70% with intelligent cache-aside patterns and implementing rate limiting middleware to prevent abuse
- Established comprehensive testing and monitoring: JUnit unit tests, MockMvc integration tests achieving 90% code coverage, and SLF4J logging framework enabling production debugging and performance analysis

### ChatPDF - ML-Powered Document Processing Infrastructure — Python, FastAPI, LangChain, PostgreSQL, Docker

[GitHub](#)

- Architected document processing backend using Python FastAPI and LangChain framework, implementing PDF parsing pipelines with PostgreSQL pgvector for semantic search processing 100+ documents with 95% retrieval accuracy
- Resolved performance bottlenecks through async/await patterns and document chunking strategies, handling files up to 50MB with optimized memory management and implementing exponential backoff retry logic for robust error handling