

ABHISHEK JANAKKUMAR JANI

Software Engineer

California, USA | +1 408-797-0796 | abhishekjanidev@gmail.com | [LinkedIn](#) | [GitHub](#) | [Portfolio](#)

SUMMARY

Software Engineer with 3+ years of experience building scalable backend and AI-integrated applications across fintech and enterprise environments. Proficient in Java, Spring Boot, Python, FastAPI, REST APIs, gRPC, microservices, Kafka, Redis, and Azure. Experienced in developing secure payment integrations, MCP server services, asynchronous workflows, and enterprise search platforms. Skilled in OAuth 2.0, JWT, Docker, CI/CD, automated testing, and system monitoring, with a strong focus on application performance, security, reliability, and operational efficiency.

SKILLS

Programming Languages	Python, Java, JavaScript, C, C++, SQL, Object-Oriented Programming (OOP), TypeScript, Bash/Shell Scripting, Data Structures & Algorithms, Multithreading, Concurrency
Backend Development	Spring Boot, Spring MVC, Hibernate, FastAPI, Django, Flask, Node.js, RESTful APIs, SOAP Web Services, GraphQL, gRPC, WebSockets, Microservices, API Development, Authentication, Authorization, JWT, OAuth 2.0, Spring Security, API Versioning
Frontend Development	React.js, JavaScript, HTML, CSS, TypeScript, Responsive Web Design, AJAX, JSON, Bootstrap, Redux, React Hooks, State Management, Frontend Performance Optimization
Cloud & Infrastructure	AWS, Google Cloud Platform (BigQuery, Vertex AI), Docker, Kubernetes, Terraform, AWS Lambda, EC2, S3, IAM, Cloud Deployment, CloudFormation, VPC, CloudWatch, Auto Scaling, Load Balancers
Databases & Caching	PostgreSQL, MySQL, MongoDB, DynamoDB, Snowflake, Cassandra, Redis, Database Design, Query Optimization, SQL Optimization, Database Indexing, Transactions, Stored Procedures, Data Modeling, NoSQL Databases
System Design & Architecture	Distributed Systems, Microservices Architecture, Event-Driven Architecture, Software Design Patterns, API Gateway, Load Balancing, Caching, Kafka, RabbitMQ, Scalability, High Availability, Fault Tolerance, System Architecture, CAP Theorem, SOLID Principles, Low-Level Design (LLD), High-Level Design (HLD), Service Discovery, Circuit Breaker, Rate Limiting
AI & Machine Learning	Supervised Learning, Unsupervised Learning, Deep Learning, CNNs, RNNs, LSTMs, Transformers, NLP, Text Classification, Sentiment Analysis, Recommendation Systems, Time-Series Forecasting, Generative AI, LLMs, RAG, Prompt Engineering, Model Evaluation, Feature Engineering, ML Pipelines, MLOps, Model Deployment, Vector Databases, Embeddings, Fine-Tuning, Hugging Face, LangChain, LangGraph
Testing & API Tools	JUnit, Mockito, pytest, unittest, Postman, Swagger/OpenAPI, Code Review, Security Best Practices, Unit Testing, Integration Testing, API Testing, Test Automation, End-to-End Testing, Test-Driven Development (TDD), Performance Testing, SonarQube
DevOps & CI/CD	Git, GitHub, GitLab, Jenkins, CircleCI, CI/CD Pipelines, Version Control, Build Automation, Deployment Automation, Containerization, Infrastructure as Code (IaC), GitHub Actions, Maven, Gradle, Linux, Shell Scripting, Monitoring, Logging, Prometheus, Grafana
Methodologies & Collaboration	Agile, Scrum, SDLC, Jira, Confluence, Sprint Planning, Code Review, Technical Documentation, Team Collaboration, Requirements Analysis, Software Development Life Cycle, Problem Solving, Cross-Functional Collaboration

PROFESSIONAL EXPERIENCE

PayPal, CA, USA

Sep 2025 - Present

Software Engineer - AI integration

- Designed schema-driven Java modules integrating PayPal's Checkout, Merchant, and Invoicing APIs into an internal Agent Toolkit, reducing development effort for new agent capabilities by approximately 25% across participating engineering teams.
- Built and maintained scalable MCP server services using Java, Spring Boot, and gRPC, enabling AI agents to securely access approved payment, merchant, and invoicing functions in real time.
- Implemented OAuth 2.0, JWT validation, scoped tool permissions, and rate limiting to protect agent-facing services and enforce secure access to sensitive financial APIs across internal environments.
- Created asynchronous Python workflow components using asyncio and aiohttp for invoice creation, tax calculations, and discount validation, reducing average workflow completion time by 22% through optimized task execution.
- Supported event-driven agent workflows using Kafka and Redis Streams, adding state tracking, automated retries, and failure-handling mechanisms for dependable multi-step processing across distributed services.

- Improved application performance through non-blocking I/O, gRPC streaming, connection pooling, and back-pressure handling, supporting 25K–35K agent calls per hour with median latency below 170 milliseconds.
- Integrated Python-based transaction-risk scoring services into agent workflows, improving approval accuracy for eligible low-risk transactions by approximately 8% while maintaining established risk controls.
- Established monitoring and logging with Prometheus, Grafana, and the ELK Stack, using performance insights to identify payload bottlenecks and reduce average response latency by 14% across critical services.
- Developed unit, integration, and contract tests using JUnit 5, Mockito, PyTest, and Newman, strengthening service reliability and reducing recurring issues during scheduled production releases.
- Collaborated with API Platform, Security, Risk, and AI/ML teams to refine tool schemas, validate API contracts, troubleshoot integrations, and support Agent Toolkit adoption across internal product teams and business units.

Deloitte, Mumbai, India

May 2022 - Jun 2024

Software Engineer

- Developed Python and FastAPI services for an internal knowledge-search platform, supporting secure access to compliance, policy, and operational documentation across multiple business teams.
- Created Python ingestion pipelines to extract, clean, segment, and tag internal documents, processing more than 60,000 files daily and storing curated content in Azure Blob Storage.
- Configured Azure Cognitive Search with keyword, metadata, and semantic-search capabilities, enabling employees to retrieve relevant information from approximately 1.5 TB of enterprise content.
- Built asynchronous REST APIs for document search, filtering, user feedback, and access management, supporting over 5,500 employee searches daily with average response times below three seconds.
- Developed the user interface using React and TypeScript, implementing conversational search, document previews, filters, and role-based views that increased monthly active usage from 280 to 950 users.
- Implemented text preprocessing, relevance-ranking, and query-expansion techniques in Python, improving successful document retrieval from 71% to 83% across internal evaluation queries.
- Integrated Azure Active Directory OAuth 2.0 and JWT-based authorization, ensuring controlled document access and maintaining auditable records of user searches and system activity.
- Containerized application services using Docker and automated CI/CD deployments through Azure DevOps, reducing average deployment time from 42 minutes to approximately 16 minutes.
- Configured application monitoring through Azure Monitor, Prometheus, and Grafana, helping identify latency and resource issues and reducing recurring production incidents by 18%.
- Collaborated with compliance, data, and product teams during Agile sprints to refine search logic, validate document access rules, resolve defects, and support phased platform releases.

PROJECTS

Mistral City - Repository Intelligence Platform | *TypeScript, Node.js, Express, React, SSE* | [Github](#) | 1st Place, Mistral Hackathon (\$3,000) | 2026

- Built a repository intelligence engine that maps semantic systems and emits evidence-backed health and security grades.
- Streamed typed analysis and agent events over SSE via a Node bridge with schema validation, caching, and a CLI harness

Amortize - Cost-Reducing LLM Gateway Proxy | *Python, FastAPI, Snowflake, SQLite, EverOS, Streamlit* | [Github](#) | 2026

- Built an LLM gateway proxy that reduced token usage through context compaction while preserving output compatibility.
- Developed reusable workflows with cost tracking, batching, retries, and recovery, achieving 97% token reduction on repeat runs.

EDUCATION

Master of Science in Computer Science

California State University, Fullerton, CA

Bachelor of Engineering, Information Technology

Atharva College of Engineering, University of Mumbai, Mumbai, India

CERTIFICATIONS

- **AWS Certified Developer – Associate**
- **Claude AI Certification – Anthropic**