

Harsh Patel

USA | +1 (315) 928 8459 | harshpatel180801@gmail.com | [LinkedIn](#) | [GitHub](#) | harsh18.com

Applied AI Software Engineer with 3+ years of experience building production AI, backend, and data-intensive systems using Python, Java, SQL, and React. Experienced developing LLM workflows, RAG services, structured-output guardrails, scalable APIs, and asynchronous processing systems. Strong focus on grounding AI responses in authoritative data, evaluating system reliability, and owning features from requirements and architecture through testing, deployment, monitoring, and production support.

SKILLS

Languages & Frontend: Python, Java, JavaScript/TypeScript, SQL, React

Applied AI: LangGraph, LangChain, LLM Agents, Tool Calling, Prompt Engineering, RAG, Context Assembly, Structured Outputs, Agent Evaluation, pgvector

Data Systems: PostgreSQL, ClickHouse, Kafka, Redis, DynamoDB, Spark/PySpark, ETL Pipelines

Backend & Production: FastAPI, Spring Boot, REST APIs, Docker, Git, CI/CD, Pytest, Monitoring, Structured Logging, OpenTelemetry

Machine Learning: PyTorch, scikit-learn, Hugging Face, Embeddings, Anomaly Detection, Model Evaluation

Cloud: AWS, Azure, GCP

EXPERIENCE

Honeywell | AI Software Engineer (Co-op)

Sep 2025 - Aug 2026

- Collaborated with operations and domain experts to map production workflows, identify authoritative data sources, and translate recurring diagnostic and configuration tasks into requirements for AI-assisted software.
- Designed a production AI agent using Python, FastAPI, LangGraph, and OpenAI, defining typed tools, context-assembly logic, prompt instructions, structured outputs, and JSON Schema guardrails for generating verifiable operational recommendations.
- Built a RAG pipeline using pgvector and approximate-nearest-neighbor retrieval to ground responses in operating procedures, configuration documentation, and historical incidents, returning supporting evidence with each recommendation.
- Integrated the agent with PostgreSQL records, ClickHouse telemetry, and Kafka event streams, applying time-window validation, idempotent processing, bounded retries, and human approval before executing state-changing actions.
- Developed automated evaluation datasets and graders for answer correctness, source grounding, tool selection, and unsafe-action prevention, running regression evaluations through CI/CD before deployment.
- Delivered a React interface and production monitoring for inspecting agent evidence, approving actions, and tracking latency, tool failures, retrieval quality, and model behavior through structured logs and distributed traces.

IConsult | Machine Learning Engineer

Jun 2025 - Aug 2025

- Built a personalized clothing recommendation engine using Python, collaborative filtering, and user interaction data to rank products based on swipes, likes, categories, brands, and browsing behavior, achieving 78% Precision@10.
- Developed data pipelines to transform user-product interactions into recommendation features and generate real-time suggestions, increasing recommendation click-through rate by 24% and integrated the ML engine with the marketplace application through REST APIs and implemented cold-start recommendations using product similarity and popularity-based ranking.

MindTree | AI Software Engineer (Intern to Co Op)

Jan 2022 - Jul 2024

- Developed an NLP-based workflow intelligence service using Python, spaCy, and transformer models to classify unstructured enterprise requests, extract relevant fields, and recommend approval routes and task assignments.
- Built the reliability harness around the model, including text preprocessing, structured response schemas, confidence thresholds, validation rules, rule-based fallbacks, and human review for low-confidence predictions.
- Created labeled evaluation datasets from historical workflow cases and automated regression tests for classification accuracy, field extraction, routing correctness, and fallback behavior to identify model-quality regressions before deployment.
- Integrated AI predictions with Java Spring Boot services and PostgreSQL workflows through versioned REST APIs, preserving role-based access controls, approval rules, and auditable records for every automated recommendation.
- Optimized Hibernate queries, database indexes, and API processing to reduce p95 latency from 1.8 seconds to 320 milliseconds while supporting more than 800 enterprise clients and 50,000 daily users.
- Built asynchronous processing workflows using Redis, idempotency keys, bounded retries, and exponential backoff for model inference, data synchronization, and notification delivery, improving pipeline reliability by 30%.

EDUCATION

Master of Science, Computer Science

May 2026

Syracuse University

PROJECT

Equipment data migration and maintenance platform

Python, FastAPI, PostgreSQL, Kafka, React, Docker

- Built a fault-tolerant migration service for equipment, maintenance, and work-order records, using bounded batch workers, checkpoint recovery, idempotent source-to-target mapping, data validation, and partial-success reporting.
- Developed versioned REST APIs and Kafka event workflows for migration jobs, equipment updates, and maintenance requests, using retry policies and dead-letter processing to prevent individual record failures from blocking an entire migration.
- Created a React operations dashboard for monitoring migration progress, investigating validation failures, retrying failed records, and managing work orders through role-based access control and auditable status transitions.
- Added an AI troubleshooting assistant that grounds migration-failure explanations in validation rules, schema documentation, and processing logs while requiring human confirmation before retrying or modifying records.

CERTIFICATION

- AWS Solution Architect, AWS Academy Machine Learning Foundations
- Azure AI Fundamentals
- SAP Certified Application Associate - SAP Analytics Cloud
- Google Cloud Learning Skills Badges