RAM GOLLADI

Location: Canada / India (Open to Relocation)

Email: ramgolladi1503@gmail.com | Phone: +1-519-614-1503

GitHub: github.com/ramgolladi | LinkedIn: linkedin.com/in/ramgolladi

PROFESSIONAL SUMMARY

QA Engineer with 4+ years of experience in enterprise testing and 1+ year building a production-grade algorithmic trading bot using Python, real-time market data, and automated risk controls. Strong background in test automation, system validation, failure analysis, and AI-driven testing. Experience working in large organizations (IBM) and independently owning end-to-end backend systems with live data, auditability, and governance. Known for identifying deep system flaws before production failures.

CORE SKILLS

Testing and QA

- Test Strategy, Test Planning, Test Case Design
- Functional, Integration, Regression, System Testing
- API Testing (REST, WebSocket)
- Defect Lifecycle Management
- Risk-Based Testing
- Production Validation and Monitoring

Automation and Tools

- Python, PyTest, Selenium, Appium
- SQL (SQLite, PostgreSQL concepts)
- Git, GitHub, CI/CD
- Linux / macOS
- Docker (basic operational usage)

Trading and Data Systems

- Algorithmic Trading Systems
- Real-time Data Processing
- Zerodha Kite API (REST + WebSocket)
- Market Data Feeds (Ticks, Depth, Snapshots)
- Risk Management Systems
- Audit Logs and Trade Governance

AI and Advanced

- ML Model Validation
- XGBoost Model Integration
- AI Testing (Bias, Drift, Data Integrity)
- ISTQB AI Testing (CT-AI aligned concepts)

PROFESSIONAL EXPERIENCE

IBM - QA Engineer

IBM India Pvt Ltd | 2019 - 2023

Role and Responsibilities

- Designed and executed functional, regression, and integration test suites for enterprise applications.
- Automated critical workflows using Python-based frameworks and Selenium.
- Validated APIs and backend services, identifying data integrity and contract issues.
- Worked closely with developers, product owners, and DevOps teams to resolve production blockers.
- Participated in root cause analysis (RCA) for production defects.
- Ensured compliance with enterprise quality standards and SLAs.

Positive Outcomes

- Reduced post-release defects by 30%+ through improved regression coverage.
- Identified critical data consistency issues before production rollout.
- Improved test execution time via automation adoption.

Negative Outcomes and Learnings

- Over-reliance on manual regression in early phases caused release delays.
- Fixed by introducing risk-based prioritization and automation for high-impact flows.

How Achieved

- Shifted focus from "test everything" to "test what can break revenue or trust."
- Built reusable automation components instead of one-off scripts.

INDEPENDENT PROJECT EXPERIENCE

Algorithmic Trading Bot (Python) - End-to-End System Owner

2025 - Present (1+ Year Ongoing)

Project Overview

Designed and built a production-grade algorithmic trading bot for Indian index markets using real-time data, automated decision logic, and strict governance controls. System designed to operate safely under live market conditions with full auditability.

Key Responsibilities

- Designed system architecture covering market data ingestion (REST + WebSocket), strategy evaluation, risk management, trade execution, and audit/reporting.
- Integrated Zerodha Kite API for live trading and paper trading.
- Built SQLite (WAL mode) database for trades, ticks, and depth snapshots.
- Implemented deterministic trade IDs, full audit logs, and decision hashes for replay and forensic analysis.
- Created pre-market regression suites to validate authentication, market feeds, strategy invariants, and risk limits.
- Designed kill-switches and fail-closed behavior to prevent runaway trades.

Positive Outcomes

- Achieved stable real-time data streaming with controlled resource usage.
- Eliminated silent failures through strict validation and logging.
- Built a system that can be replayed, audited, and debugged deterministically.
- Demonstrated production-level thinking beyond basic strategy scripts.

Negative Outcomes (Critical and Honest)

- Initial versions suffered from SQLite file descriptor leaks, WebSocket lifecycle mismanagement, and schema drift in trade logs.
- These issues caused instability under load.

How Issues Were Solved

- Introduced connection lifecycle controls, resource leak detection, schema enforcement for trades, and regression gates before market open.
- Treated failures as system design flaws, not bugs to patch.

Why This Project Matters

This is not a toy bot. It demonstrates backend engineering discipline, QA mindset applied to live systems, risk-aware system design, and production-grade observability.

KEY ACHIEVEMENTS

- Built and stabilized a live system processing real-time financial data.
- Applied QA principles to algorithmic trading, reducing hidden failure modes.
- Demonstrated ownership of architecture, testing, deployment, and monitoring.
- Transitioned from tester mindset to system reliability engineer mindset.

EDUCATION

Bachelor's Degree (Details omitted for ATS neutrality - add if required)

CERTIFICATIONS AND LEARNING

- ISTQB Foundation (Conceptual Alignment)
- ISTQB AI Testing (CT-AI) - In Progress
- Continuous self-learning in ML Testing, AI Risk and Drift, and Trading Systems