

// GENAI/ML BOOTCAMP CAPSTONE

# AUTONOMOUS INCIDENT RESPONSE PLATFORM

Automating outage remediation caused by severe compute overages, built through the lens of a high-volume e-commerce company hit by sudden traffic surges like unexpected product drops.

A force multiplier for on-call IT and SRE teams fighting late-night outages.

// THE PROBLEM & PIPELINE

# FROM 2 AM PAGER TO FIXED IN SECONDS

Manually, engineers wake up, parse thousands of raw logs, look up runbooks and type terminal commands, so MTTR stays high.

01

## Parse & Detect

Telemetry ingested via spaCy; ML isolates spikes like CPU > 85%.

02

## Retrieve Runbooks (RAG)

FAISS vector search pulls the right mitigation protocol.

03

## Execute Remediation

Local llama3.1 calls FastMCP tools to run the fixes autonomously.

// SOLUTION & BENEFITS

# EXAMPLE: CPU SPIKES PAST 85%

Cause: rogue background threads or unindexed queries. Instead of a human troubleshooting, FastMCP tools run the runbook.

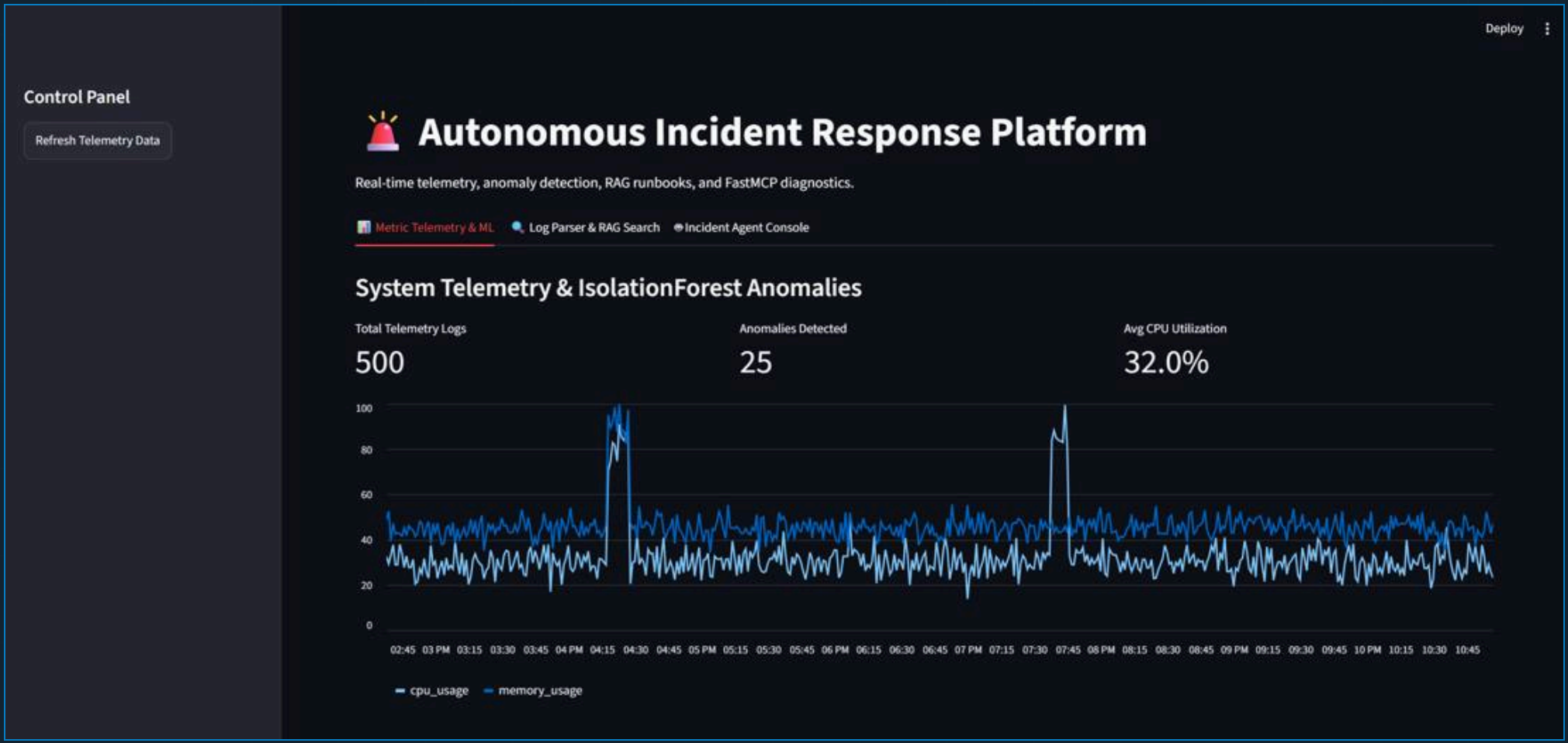
|                                               |                                        |                                     |
|-----------------------------------------------|----------------------------------------|-------------------------------------|
| <b>Inspect</b><br>Isolated process inspection | <b>Terminate</b><br>Kill rogue threads | <b>Verify</b><br>Index verification |
|-----------------------------------------------|----------------------------------------|-------------------------------------|

# HOURS OF DOWNTIME → SECONDS

Who benefits: SREs, DevOps and on-call engineers in E-commerce, Fintech, Healthcare and SaaS. Less alert fatigue, faster MTTR, fully local and private via Ollama, zero cloud API cost.

// LIVE DEMO

# INTERACTIVE STREAMLIT DASHBOARD



Synthetic anomaly engine

Real-time telemetry parsing

Local RAG retrieval

Autonomous fix execution

Live remediation tracking

[View code on GitHub ↗](#)

[▶ Watch Loom demo](#)

// TECH STACK

# BUILT ENTIRELY WITH LOCAL, OPEN TOOLING

- Python
- Streamlit
- FastMCP (MCP)
- Ollama · llama3.1
- FAISS
- RAG
- spaCy
- Docker & Compose
- Git & GitHub

## Challenge

No production-grade, real server logs with tagged incidents for training and testing.

## Solution

An automated synthetic telemetry pipeline that simulates CPU spikes, database locks and memory leaks with full control.

// FUTURE STEPS & CONCLUSION

# WHAT'S NEXT: ONE PLATFORM, MANY INDUSTRIES

Making the system adaptable to E-commerce, Fintech, Healthcare and SaaS, each with its own tools and detailed runbooks.

### Flexible model routing

Toggle between local Ollama and Claude/GPT-4. Keep data private and API cost low with Ollama; flip the flag when a task needs deeper reasoning.

### Deeper runbooks

Expand FAISS-indexed runbooks with industry-specific outage causes and a wide range of detailed, edge-case instructions.

### Expanded tool library

Add tested network, database and container fixes to FastMCP, executed only after runbook instructions are confirmed.

Achieved: automated log parsing, anomaly detection, vector runbook retrieval and FastMCP tool execution.

**REACTIVE FIREFIGHTING → PROACTIVE REMEDIATION, SECURE, LOCAL AND COST-EFFECTIVE.**

// CLOSING //

# THANK YOU

Autonomous Incident Response Platform

GABRIEL T. KESSLER · GENAI/ML BOOTCAMP CAPSTONE



SCAN FOR THE CODE ON GITHUB

Questions welcome.