AgentOps - Four-Pillar Operating Checklist

Fora Soft Learn · AI for Video Engineering

1 - Observability (you can't run what you can't see)

```
Trace = one whole job;  span = one step (an LLM call or a tool call) with tokens + latency.
One user interaction is commonly 40-75 spans. Log only the final answer and you can't replay a failure.
OpenTelemetry GenAI: invoke_agent -> chat + execute_tool spans;  gen_ai.usage.input_tokens / output_tokens.
```

2 - Evaluation (is it actually good? - the pillar teams skip)

```
Golden set     fixed tasks with known-good outcomes; re-run on every model / prompt / tool change.
Three levels   end-to-end (did it succeed) - trajectory (was the path sound) - component (which tool broke).
LLM-as-judge   a stronger model scores against a rubric; a 0-5 scale aligns best with humans (~0.89).
Reliability    measure pass^k, not just pass^1: 0.9 per step = 0.9^8 = 43% over eight steps.
```

3 - Cost (measure dollars per SUCCESSFUL task, not per token)

```
An agent loops: 3-8 model calls per task; 50k-200k tokens for one 'simple' task; calls = 70-85% of cost.
Example: 5 calls x 30,000 tokens = 150,000;  x $5/M = $0.75/task;  x 10,000/day = $7,500/day.
A $0.75 agent that fails half the time and retries costs $1.50 per success. Route / cache / batch to cut it.
```

4 - Security (an agent ACTS, so a compromise is a bad action, not a bad sentence)

```
Reference: OWASP Top 10 for Agentic Applications (ASI01-ASI10, Dec 2025) + NIST AI RMF GenAI Profile.
Top risks for a video agent: goal hijack (ASI01), tool misuse (ASI02), memory poisoning (ASI06).
Signature attack: INDIRECT prompt injection - a command hidden in a caption / transcript the agent reads.
Defenses: input guardrails (data != instructions) + least-agency permissions + human gate on big actions.
```

5 - Before you ship the agent, ask:

- [ ] Is every agent run traced (each LLM and tool call as a span), so a bad output can be replayed, not guessed?
- [ ] Is there a golden set the agent is scored against on every change - at task, trajectory, and component level?
- [ ] Is reliability reported as pass^k (repeat-success), not just average pass^1 'it worked when I tried it'?
- [ ] Is cost tracked as dollars-per-successful-task, with easy tasks routed to a cheaper model and context cached?
- [ ] Does the agent hold least-agency permissions - the minimum tools and scopes, bound to the specific user?
- [ ] Are input guardrails in place so a command hidden in a caption/transcript can't become an instruction?
- [ ] Does a human approve every expensive, irreversible, or high-stakes action before the agent takes it?
- [ ] Can the team show its controls against a named framework (OWASP ASI / NIST AI RMF / EU AI Act)?