The Agent Acceptance Checklist

Use this before approving any AI agent for pilot, production, vendor purchase, or workflow expansion.

1. Responsibility Boundary

- ■ The agent has one clear job.
- ■ Allowed actions are written down.
- ■ Forbidden actions are written down.
- ■ Explicit escalation triggers exist.
- ■ A human owner is accountable for behavior.

2. Workflow Fit

- ■ Workflow is mapped from trigger to completion.
- ■ Every handoff is visible.
- ■ Every tool call is named.
- ■ Every data source is named.
- ■ Customer-impacting and regulated-data moments are marked.

3. Failure Classes

- ■ Hallucination or unsupported claim.
- ■ Tool misuse or silent tool failure.
- ■ Policy / compliance breach.
- ■ Escalation miss.
- ■ Audit gap or non-reconstructable decision.

4. Adversarial Scenarios

- ■ Normal, confused, angry, and missing-information users.
- ■ Out-of-scope requests and instruction override attempts.
- ■ Sensitive-data requests and regulated workflow triggers.
- ■ Tool outage and incorrect tool result.
- ■ Policy conflict and required escalation.

5. Tool-Call Safety

- ■ Explains material actions before taking them.
- ■ Gets confirmation before irreversible actions.
- ■ Recognizes failed tool calls.
- ■ Does not claim completion when a tool failed.
- ■ Tool permissions are scoped to the workflow.
- ■ Logs show what was called, when, why, and with what result.

6. Human-in-the-Loop

- ■ The agent knows when to stop.
- ■ Escalation handoff includes full context.
- ■ Human reviewer can see source evidence.
- ■ Clear approval gate for consequential actions.
- ■ There is a kill switch.

Critical No-Go Triggers

- ! Fabricates policy, price, eligibility, legal, financial, or medical advice.
- ! Executes a material action without required confirmation.
- ! Fails to escalate distress, threat, fraud, complaint, protected-class, or regulated-topic scenarios.
- ! Cannot produce an audit trail for consequential actions.

- ! Leaks or mishandles sensitive data.
- ! Ignores explicit workflow boundaries.
- ! Claims completion when a tool call failed.

If this agent failed publicly, could we prove we tested the failure mode before launch?