KubeCon | CloudNativeCon
North America 2026

Cadence Workflow ContribFest: Build Durable AI Agent Examples

Salt Lake City, Utah, United States





Diana Zawadzki Shijie Sheng

Make your agent recover from a failed call

Before



One failed call.
No draft.

After



Retry the lookup.
Reach human review.

Your workshop guide

Setup, exercise and tasks

github.com/cadence-workflow/cadence-ai-samples/tree/main/contribfest/kubecon-na-2026/



In your checkout:
contribfest/kubecon-na-2026/

Your local setup and connections

Docker + Compose. Python 3.12 / 3.13. Ollama.



Docker starts Cadence, Cassandra and Web together.

Ollama: localhost:11434 **Web: localhost:8088**

Your recipe connects to Cadence. Cadence manages storage.

Start Cadence and check its services

Start Docker. Registration requires the Cadence CLI.

```
git clone https://github.com/cadence-workflow/cadence.git
cd cadence
docker compose -f docker/docker-compose.yml up -d
docker compose -f docker/docker-compose.yml ps
```

Wait for Cassandra healthy, Cadence and Web running.

```
cadence --domain cadence-ai-samples domain register
```

Already running? Check and reuse your local stack.

Prepare the sample on your laptop

Fork cadence-ai-samples on GitHub. Replace YOUR-USERNAME.

```
git clone https://github.com/YOUR-USERNAME/cadence-ai-samples.git
cd cadence-ai-samples
ollama pull qwen2.5:7b
cd recipes/incident-triage/python/bare
./run.sh setup
.venv/bin/python -m unittest discover -s tests
```

Keep Docker, Cadence and Ollama running.

The incident-triage recipe you will run

A checkout incident gives our agent a trace.

Collect evidence → **Investigate** → **Draft report** → **Human review**



Review the draft before saving it.

Synthetic incident. Real model calls in live mode.

1. Run the baseline

```
.venv/bin/python \
  exercises/retry_policy_live/run_exercise.py \
  --scenario baseline
```

Open the printed Run link.
Find the draft awaiting review.

Type reject to finish.

2. A temporary error stops the investigation

```
.venv/bin/python \
  exercises/retry_policy_live/run_exercise.py \
  --scenario rate-limit
```

Graph saved **429** **0 model calls**

Too many requests. Retry-After: 1 second.

Make the lookup recover, then verify it

Edit exercises/retry_policy_live/exercise_options.py

SCAN_ACTIVITY: wait ≥ 1s, bound attempts,
exclude NonRetryableHttpError.

```
.venv/bin/python \
  exercises/retry_policy_live/run_exercise.py \
  --scenario rate-limit
```

429, then 200. Graph runs once. Draft reaches review.

Also run --scenario missing-resource. Expect one 404.

Our live Qwen run: before and after

Observed result	Starter	With retry policy
Gateway attempts	429, stop	429, then 200
Graph lookups	1	1
Model responses	0	4
Model tokens	0	2,463

Qwen 2.5 7B. Input + output tokens. Synthetic incident.

Your first durable agent

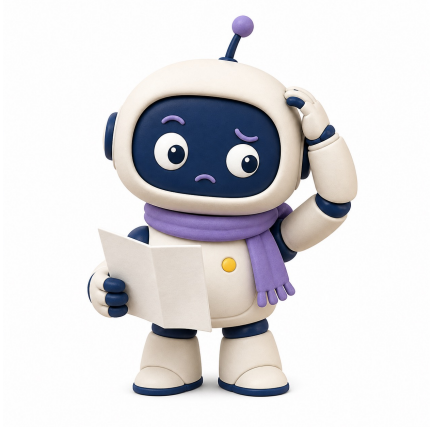


You recovered a failed lookup.

Open Cadence Web.
Find your retry and draft.

A round of applause, then keep building.

Choose what you want to build



Explore a recipe

Run a prepared agent. Inspect its history.

Improve a recipe

Choose a ready issue from the live pool.

Bring your own agent

Pick one output and one failure to recover from.

Experiment locally. Propose your idea before contributing upstream.

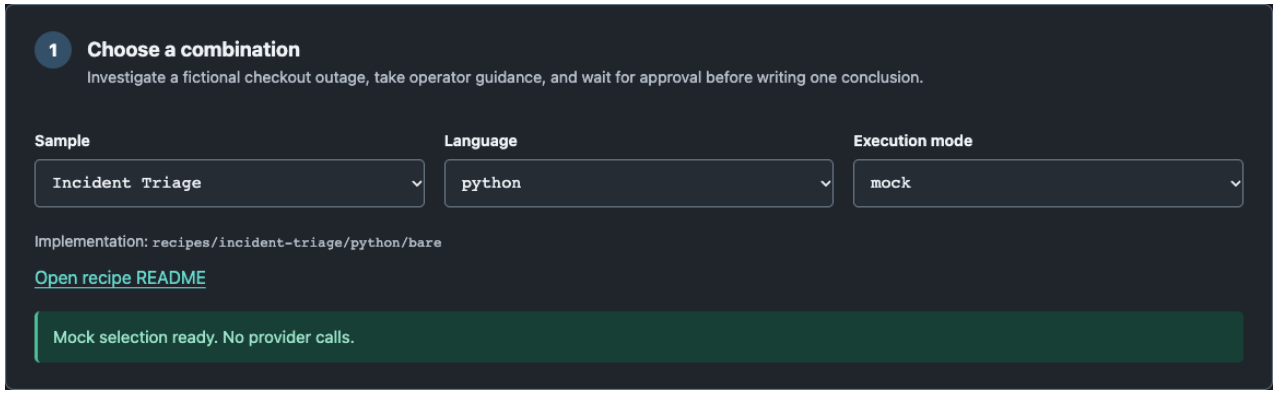
Choose a recipe you want to extend

Incident Triage	Review an incident report.
Human in the Loop	Approve a ride booking.
Agent Handoffs	Delegate a trip to specialists.
Auto Research	Review and publish a briefing.
Recurring AI Watch	Check release notes on a timer.
Ticket Routing	Assign a ticket and await acknowledgment.

github.com/cadence-workflow/cadence-ai-samples/tree/main/recipes

Explorer helps you choose matching commands

```
./scripts/explore.sh
```



Incident Triage
Python
mock

Cadence still required

A review loop for your own agent



revise: draft again with existing evidence

Reject or deadline: finish without saving.

cadenceworkflow.io/docs/python-client/signals/

Your agent as a Cadence recipe

From the samples repository root, after approval and claim:

```
./scripts/new-recipe.sh pr-review \
  --language python --classifier none \
  --agent openai-agents
```

Keep your prompts. Move model and tool I/O into Activities.

Interrupt after a recorded step.
Restart the Worker. Inspect what it reuses.

github.com/cadence-workflow/cadence-python-client



Choose your contribution by its payoff

Beginner

#139

Keep an uncertain ticket unrouted.

issues/139

Medium

#128

Expire unanswered ride approval.

issues/128

Advanced

#123

Reuse a receipt after a lost booking reply.

issues/123

github.com/cadence-workflow/cadence-ai-samples/issues

Examples only. Check the live issue before claiming.

Find and claim work in the live pool

github.com/cadence-workflow/cadence-ai-samples/issues

Filter: event:contribfest

```
/claim
```

Wait for the winning claim and status:in-progress.

Run prepared workflows locally without claiming.

Contribute your agent as a runnable example

1. Propose

One user, useful output and recoverable failure.

2. Agree and claim

Maintainer approves. /claim wins. Start your branch.

3. Build and share

Add mock data, tests and setup. Submit a draft PR with signed-off commits.

github.com/cadence-workflow/cadence-ai-samples/issues/new/choose
Choose Recipe proposal. Need help? Ask an in-room maintainer.

Common setup failures

Docker unavailable	Open Docker. Complete required sign-in.
Cadence unreachable	Check containers and port 7833.
Cassandra unhealthy	Inspect logs, Docker memory and port 9042.
Package install 401 / 403	Use the guide’s public-index setup.

Common run failures

Workflow stays pending	Match domain and Worker task list.
Query times out	Keep the matching Worker running.
Explorer says “no run”	Use the printed Workflow ID and Run ID.
Model call fails	Start Ollama. Check the selected model.

Show your result and leave a useful handoff

Show the output or Run history.

Link your issue, fork and branch.
Include commands and observed results.

Leave a draft PR or one clear next step.

Keep building with us

Workshop guide

Cadence Workflow ContribFest: Build Durable AI Agent Examples

Session feedback

#cadence-contributors on CNCF Slack

slack.cncf.io

github.com/cadence-workflow/cadence-ai-samples