TESTING ASSEMBLY

TRACK 3 – Carl Gustav

10:30–11:00

Self-Healing Test Pipelines with Multi-Agent AI

Tests from every Jira story and a pipeline that heals itself back to green

Sonal Yadav · Qaraton

Sunny Bharne · OP Pohjola

Every red build starts with the same question

Is the test wrong… or is the app wrong?

Test issue

- Expected value drifted
- Wrong test data or assertion
- App still meets the requirement

Fix the test

App defect

- A code change broke a rule
- The test is still right
- The requirement is not met

Raise a bug

Today a person triages every failure. What if agents did the triage, and the fix?

THE SYSTEM UNDER TEST

Meet the app: Bistro Order API

GET	`/health`
GET	`/menu`
GET	`/menu/{id}`
POST	`/orders/quote`

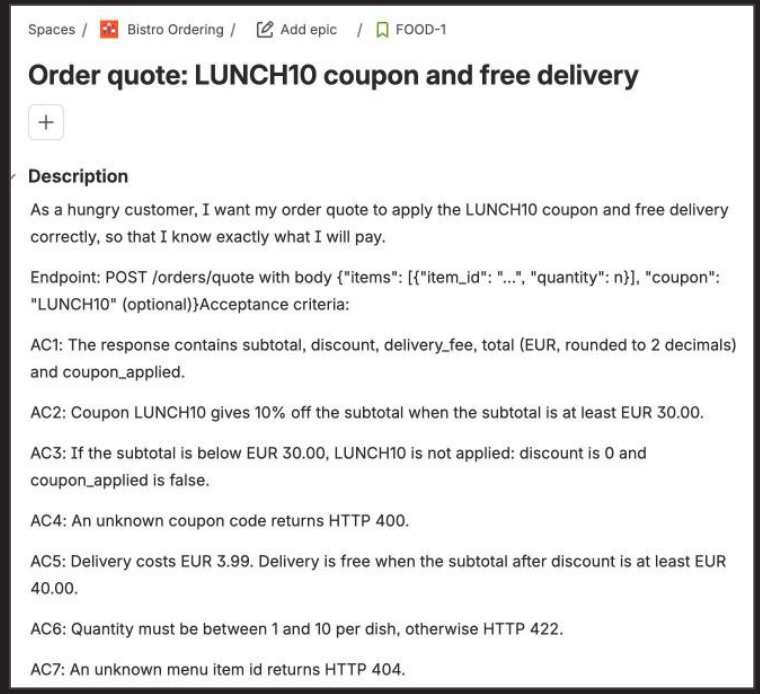
Business rules in the quote

LUNCH10
10% off from €30

Delivery
€3.99 · free from €40

Quantity
1–10 per dish, else 422

Unknown coupon
HTTP 400



Spaces / Bistro Ordering / Add epic / FOOD-1

Order quote: LUNCH10 coupon and free delivery

Description

As a hungry customer, I want my order quote to apply the LUNCH10 coupon and free delivery correctly, so that I know exactly what I will pay.

Endpoint: POST /orders/quote with body {"items": [{"item_id": "...", "quantity": n}], "coupon": "LUNCH10" (optional)}Acceptance criteria:

AC1: The response contains subtotal, discount, delivery_fee, total (EUR, rounded to 2 decimals) and coupon_applied.

AC2: Coupon LUNCH10 gives 10% off the subtotal when the subtotal is at least EUR 30.00.

AC3: If the subtotal is below EUR 30.00, LUNCH10 is not applied: discount is 0 and coupon_applied is false.

AC4: An unknown coupon code returns HTTP 400.

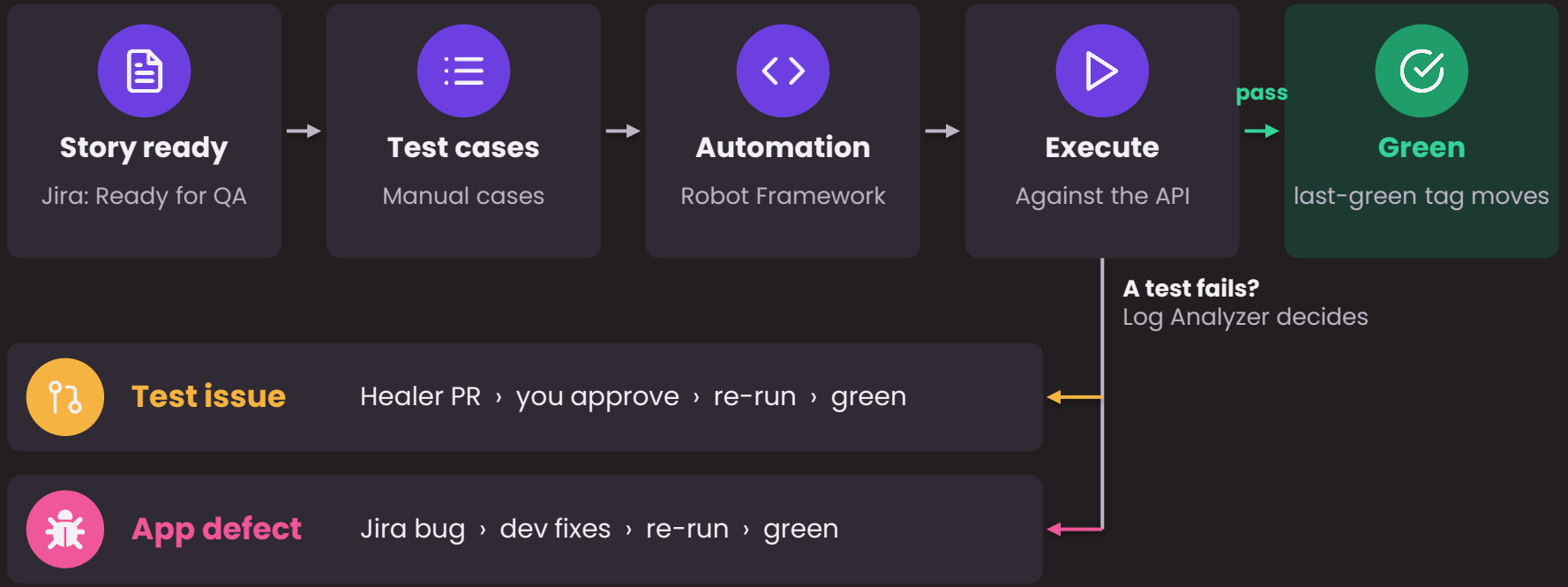
AC5: Delivery costs EUR 3.99. Delivery is free when the subtotal after discount is at least EUR 40.00.

AC6: Quantity must be between 1 and 10 per dish, otherwise HTTP 422.

AC7: An unknown menu item id returns HTTP 404.

Jira story FOOD-1: the requirement with 7 acceptance criteria

The goal: a pipeline that stays green



Self-healing is the safety net. The goal is green, with every requirement covered by tests.

Architecture: three platforms, one pipeline

Jira Cloud

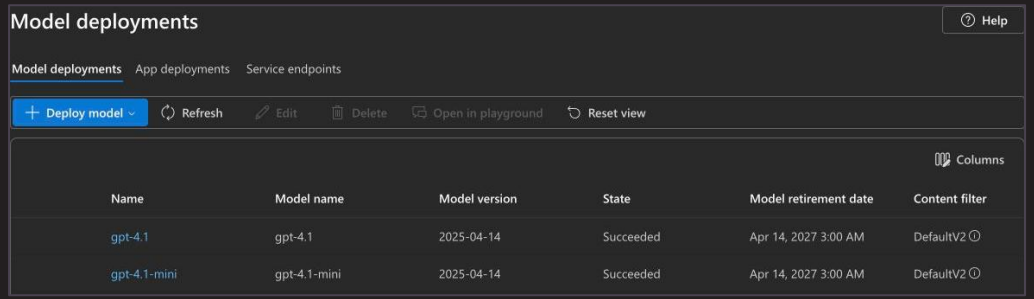
- Story with acceptance criteria
- Automation flow on Ready for QA
- Agent comments + auto bugs

GitHub Actions

- Orchestrates every step
- Runs Robot tests on the runner
- Healer PRs, approval, re-run
- Secrets: API key, tokens

Azure OpenAI

- gpt-4.1: Test Designer
- gpt-4.1-mini: other agents
- JSON mode, temp. 0
- Tokens + cost logged



Azure AI Foundry, Sweden Central

- Two deployments: gpt-4.1 and gpt-4.1-mini
- Key in GitHub Secrets; endpoint and deployment names as variables

Four agents, one job each

Test Designer

gpt-4.1

Story → 3 manual test cases

✓ Every number checked by a calculator

Automation Engineer

gpt-4.1-mini

Test cases → Robot Framework

✓ Must pass a Robot dry run

Log Analyzer

gpt-4.1-mini

Failure → verdict + root cause

✓ Requirement is the judge

Healer

gpt-4.1-mini

Test issue → minimal patch

✓ Re-runs the fix before a PR

Jira bug creation is plain code, no AI: the analyzer's findings go straight into a linked Jira bug.

Live demo in three acts

Act 1

1

New requirement

Story → Ready for QA → test cases, Robot tests, green run

Act 2

2

A test breaks

Log Analyzer: test issue → Healer PR → I approve → green

Act 3

3

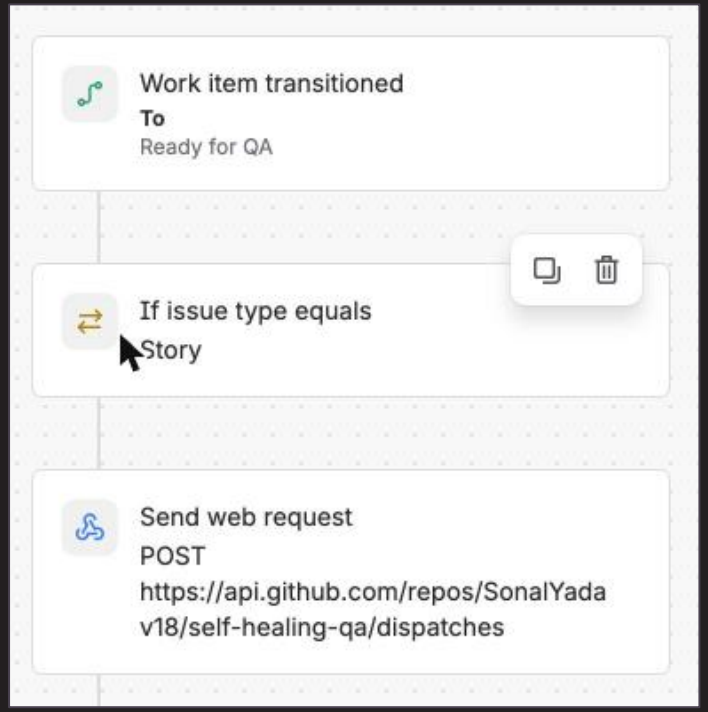
The app breaks

Log Analyzer: app defect → Jira bug with the guilty diff

Backup: the next slides show each act as screenshots from real runs

ACT 1 · TRIGGER

Moving the story to Ready for QA starts everything



Dev finishes the story

and moves it to Ready for QA

Jira Automation fires

only for issue type Story

Web request to GitHub

repository_dispatch with the issue key

Workflow 1 starts

the agents read the story from Jira

Test Designer: three manual test cases, numbers verified

🧠 Test Designer — 3 manual test cases for FOOD-1

Order quote: LUNCH10 coupon and free delivery

ID	Title	Type	Expected
TC-01	LUNCH10 coupon applied and free delivery for Pasta Carbonara x4 (subtotal >= 40 after discount)	positive	Response contains: subtotal: 56.00, discount: 5.60, delivery_fee: 0.00, total: 50.40, coupon_applied: true
TC-02	LUNCH10 coupon not applied and delivery charged for Chicken Burger x2 (subtotal just below 30)	boundary	Response contains: subtotal: 22.00, discount: 0.00, delivery_fee: 3.99, total: 25.99, coupon_applied: false
TC-03	Unknown coupon code returns HTTP 400	negative	HTTP 400 returned

💰 Token usage

Agent	Model	Input tokens	Output tokens	Total	Time	Est. cost
Test Designer	gpt-4.1	899	1,093	1,992	9.3s	$0.0105
Test Designer	gpt-4.1	974	1,139	2,113	6.5s	$0.0111
Total		**1,873**	**2,232**	**4,105**		**$0.0216**

3

test cases: positive, boundary, negative

100%

expected values re-checked by a calculator

$0.02

for this step, 4,105 tokens on gpt-4.1

Automation Engineer: Robot Framework tests, then a green run

```
*** Test Cases ***
TC-01 Lunch10 Coupon Applied And Free Delivery For Pasta Carbonara X4 (Subtotal >= 40 After Discount)
    [Documentation]    subtotal = 4 * 14.00 = 56.0
    ...                discount = round(56.0 * 0.10, 2) = 5.6
    ...                after_discount = 56.0 - 5.6 = 50.4
    ...                delivery_fee = 0.0 (free delivery since after_discount >= 40.0)
    ...                total = 50.4 + 0.0 = 50.4
    ${quote}=    Quote Order    M400:4    coupon=LUNCH10
    Should Be Equal As Numbers    ${quote}[subtotal]    56.00
    Should Be Equal As Numbers    ${quote}[discount]    5.60
    Should Be Equal As Numbers    ${quote}[delivery_fee]    0.00
    Should Be Equal As Numbers    ${quote}[total]    50.40
    Should Be Equal    ${quote}[coupon_applied]    ${True}
```

tests/generated/FOOD-1.robot as committed by the Automation Engineer

Team keywords

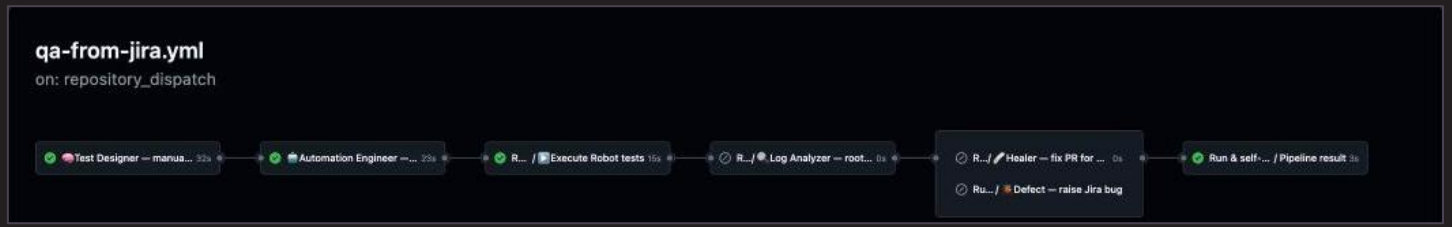
Quote Order, Get Menu Item

Validated first

Robot dry run + assertion rules

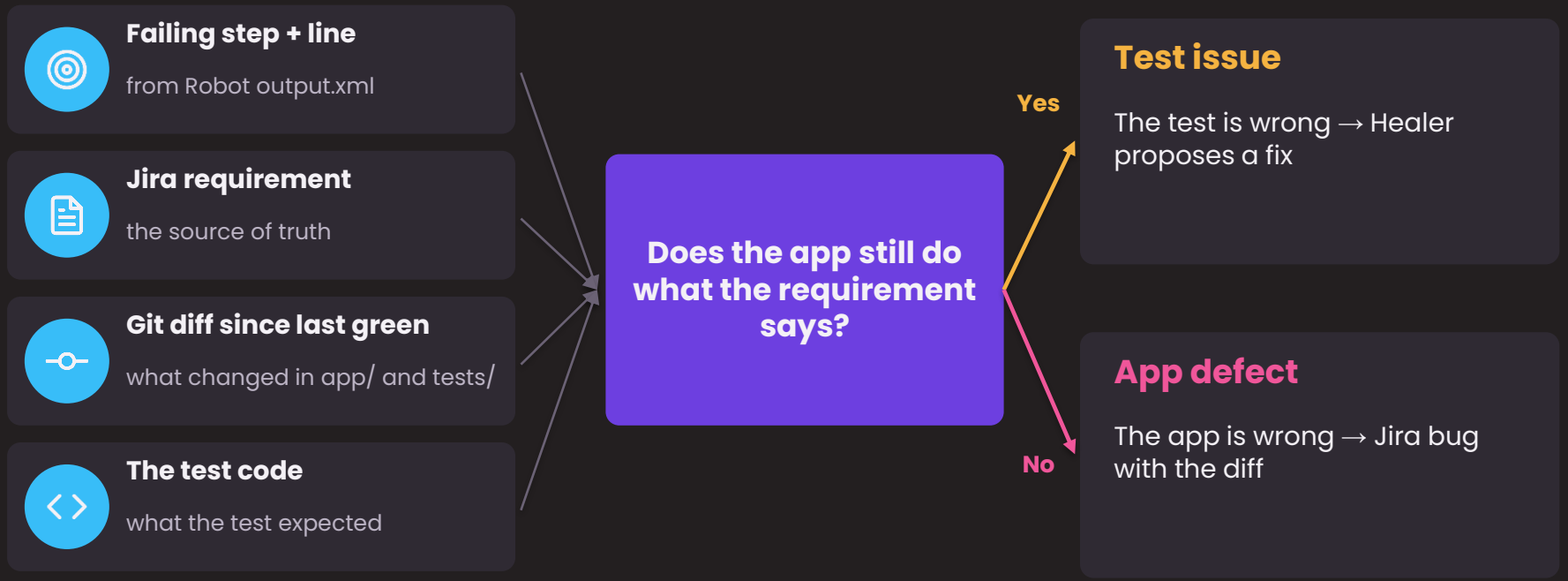
Green in 1m 24s

Jira click to finished run



WHEN A TEST FAILS

How the Log Analyzer tells a test issue from an app defect



Answer in strict JSON: verdict · root cause · evidence · remediation · confidence

READABLE BY HUMANS, NOT JUST MACHINES

The Log Analyzer's verdict, straight in the pipeline run

🔍 Log Analyzer report

⚠ Warning

The test is wrong — the app works as specified. A fix will be proposed for your approval.

Test expects discount 4.60 but requirement and app calculate 3.6 for LUNCH10 coupon on subtotal 36.0

Test	Verdict	Where	Expected	Actual
TC-01 Apply Lunch10 Coupon With Subtotal Above 30 Eur And Free Delivery Threshold Met	Test issue	`tests/generated/FOOD-1.robot:16`	discount = 4.60 (test expectation)	discount = 3.6 (requirement and app calculation)

Root cause

- Requirement FOOD-1 states LUNCH10 coupon gives 10% off subtotal >= 30 EUR
- Subtotal is 36.0 EUR, so discount should be 10% of 36.0 = 3.6 EUR
- Application code calculates discount as 3.6 correctly (line app/pricing.py:54)
- Test was changed to expect 4.60 discount, which contradicts requirement and app

Evidence

- Requirement FOOD-1 acceptance criterion 2: 'Coupon LUNCH10 gives 10% off the subtotal when the subtotal is at least EUR 30.00.'
- Test file tests/generated/FOOD-1.robot line 16: 'Should Be Equal As Numbers ${quote}[discount] 4.60'
- Test failure message: '3.6 != 4.6'
- App pricing.py lines 39-59 calculate discount as 10% of 36.0 = 3.6
- Git diff shows only test file changed, no app changes

Suspect change — `tests/generated/FOOD-1.robot:16`

💡 Tip

Remediation

- Fix test expectation to 3.6 discount as per requirement and app calculation
- Review test data and comments to ensure consistency with requirement
- Avoid changing test expected values without requirement or app changes

- Verdict, file and line, expected vs actual
- Root cause and evidence, quoting AC2
- Remediation and a confidence score
- Token usage and cost per agent

Real run: Act 2, the test-issue case (GitHub run summary)

Test issue: the Healer opens a PR, I approve it

- Patch only the failing line
- Re-run the patched test
- Open PR with evidence
- **Human reviews + approves**
- Auto-merge, re-run: green



App defect: a Jira bug, with the code change that caused it

Spaces / Bistro Ordering / Add epic / FOOD-2

[FOOD-1] LUNCH10 discount is 15% instead of 10%

Description

Raised automatically by the AI Log Analyzer agent

Requirement: FOOD-1: Order quote: LUNCH10 coupon and free delivery Ready for QA
Failing build: c094d499 — https://github.com/SonalYadav18/self-healing-qa/commit/c094d499a23cf8ce12d3b418f378937942b6112a
Pipeline run: https://github.com/SonalYadav18/self-healing-qa/actions/runs/41086165950

Failing tests

Test	Location	Expected	Actual
TC-01 Apply Lunch10 Coupon With Subtotal Above 30 Eur And Free Delivery	tests/generated/ FOOD-1: Order quote: LUNCH10 coupon and free delivery Ready for QA .robot:16	discount = 42.0 * 0.10 = 4.2 (10% off as per requirement AC2)	discount = 42.0 * 0.15 = 6.3 (15% off as per app/pricing.py line 14)

Root cause

- Requirement AC2 states LUNCH10 coupon gives 10% off subtotal >= 30 EUR
- Application code changed coupon discount from 10% to 15% in app/pricing.py line 14
- Test expects 10% discount (4.2 EUR) but app returns 15% discount (6.3 EUR)

The change that caused it

app/pricing.py

```
@@ -11,7 +11,7 @@
11  11
12  12      COUPONS = {
13  13          # code: (percent off, minimum subtotal)
14      -       "LUNCH10": (Decimal("10"), Decimal("30.00")),
    14  +       "LUNCH10": (Decimal("15"), Decimal("30.00")),
```

Commit c094d49 “feat: happy hour promo tweak” on GitHub

- Requirement AC2 says 10% → the test is right
- Expected €4.20 discount, the app gave €6.30
- Bug FOOD-2 raised and linked to FOOD-1
- No PR: the Healer leaves correct tests alone

Guardrails: AI proposes, people decide

Requirement is the judge

Jira acceptance criteria decide every verdict, not the model's opinion

Every AI output is verified

Numbers re-calculated, tests dry-run, known-bad assertions blocked

Healer must prove its fix

Patched test re-run on the current build; still failing = PR flagged

Human in the loop

Nothing merges without a PR review and approval

Least privilege

Secrets in GitHub, fine-grained token, Healer edits failing tests only

Cost and noise caps

3 test cases, mini models, temperature 0, one run at a time

What it costs, and how fast it is

$0.024

Act 1: test design + automation (gpt-4.1 + mini)

$0.004

Act 2: log analysis + heal (gpt-4.1-mini)

1m 24s

from the Jira transition to a green pipeline

Manual vs agents, per failure

	Manual	Agents
Find what is causing the failure	20–30 min	21 s
Report it: details, evidence, Jira bug	5–10 min	18 s
Total	**25–40 min**	**39 s**

Manual: our own experience. Agents: GitHub job times from the Act 3 run (FOOD-2). A person still reviews every bug and PR.

Takeaways

1 

The requirement is the judge

Classify failures against acceptance criteria, not against yesterday's test

2 

Check the AI's work like any other code

AI predicts numbers rather than calculating them, so code re-checks every expected value. Dry runs and test runs catch broken tests and failing fixes.

3 

AI proposes, people decide

PRs and Jira bugs keep humans in control, for about $0.02 of AI per run

Stack: Jira · GitHub Actions · Azure OpenAI · Robot Framework · Python

Thank you