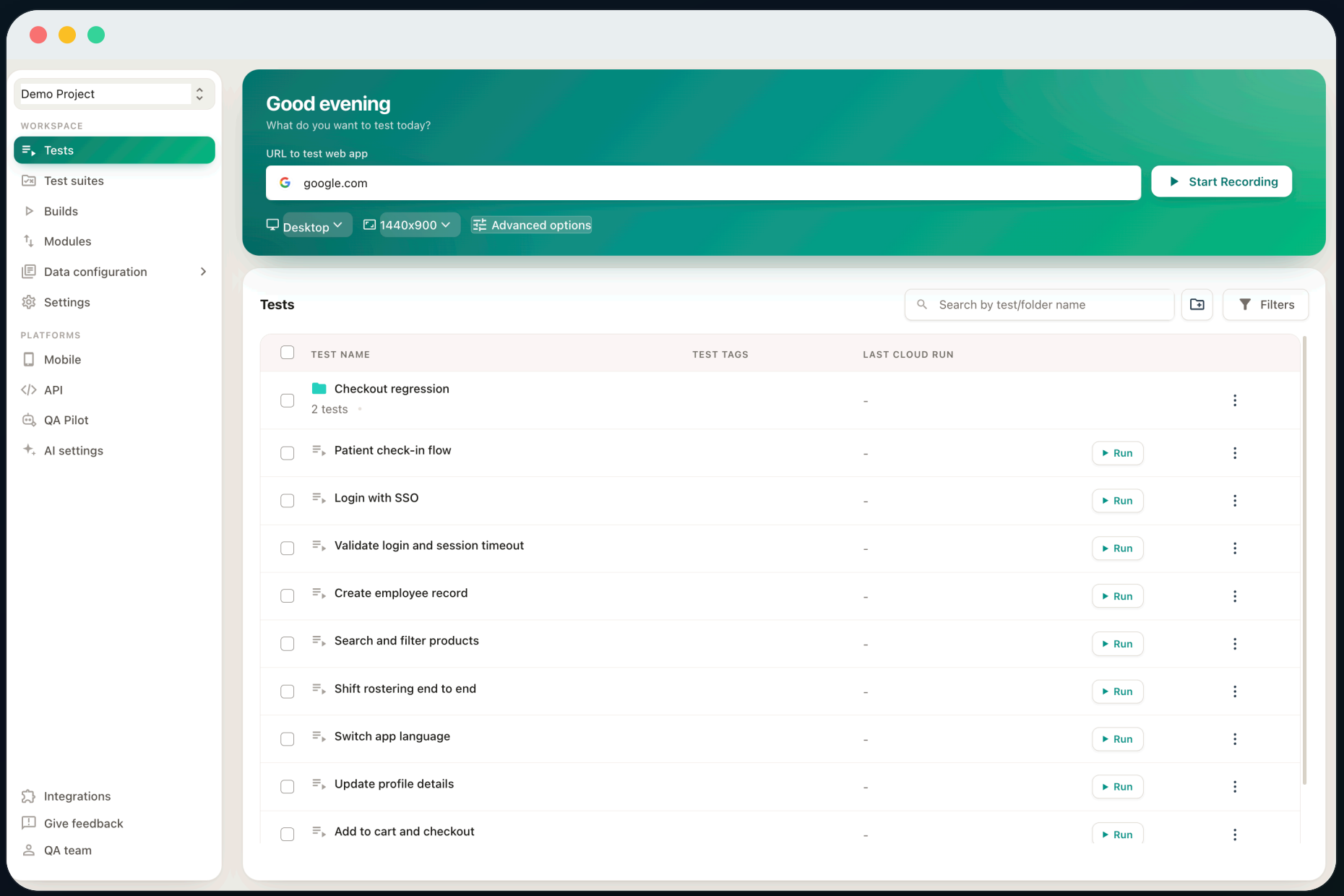
INTRODUCING

AI Automation Studio

Record, build, run and report end-to-end tests with AI. Web, mobile and API in one place, running on your own machine.

No-code recorder **AI test steps** **Runtime engine for CI** **Local-first**



THE PROBLEM

Test automation still fails people quietly.

01 **Recordings vanish.** A 50-step flow saves with zero steps, and nobody is told.

02 **Tests pass without testing.** A click lands on the wrong layer and the step never really happens.

03 **One journey, four tools.** Web, mobile, API and database checks live in separate products.

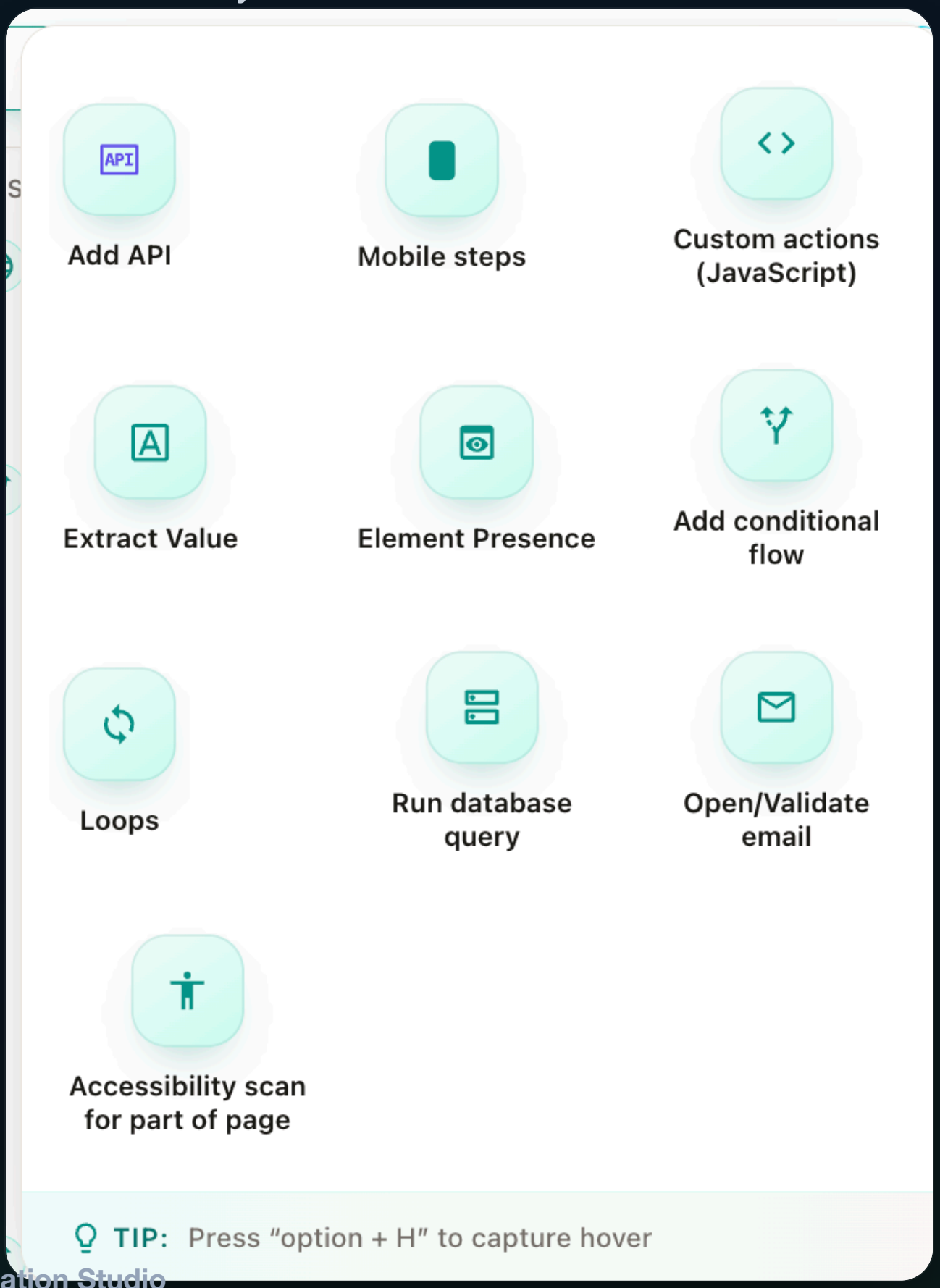
04 **Accessibility comes last.** It is audited after release, by someone else.

05 **Results stay in the tool.** They never reach the board, the channel or the pipeline.

EVERY ACTION, ONE GLANCE

Ten building blocks, one tap away.

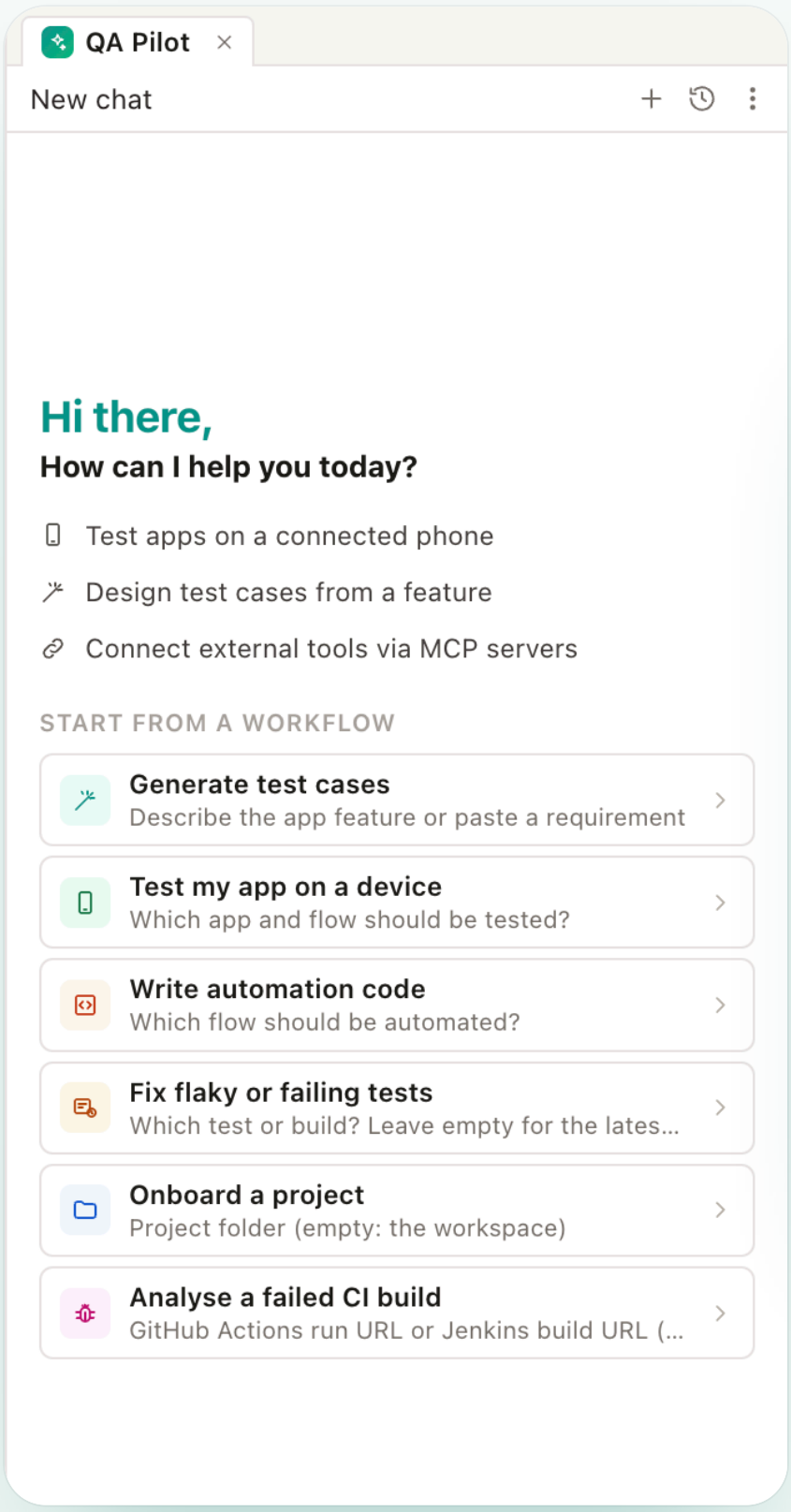
API calls, mobile steps, custom JavaScript, value extraction, element checks, conditions, loops, database queries, email checks and partial accessibility scans.

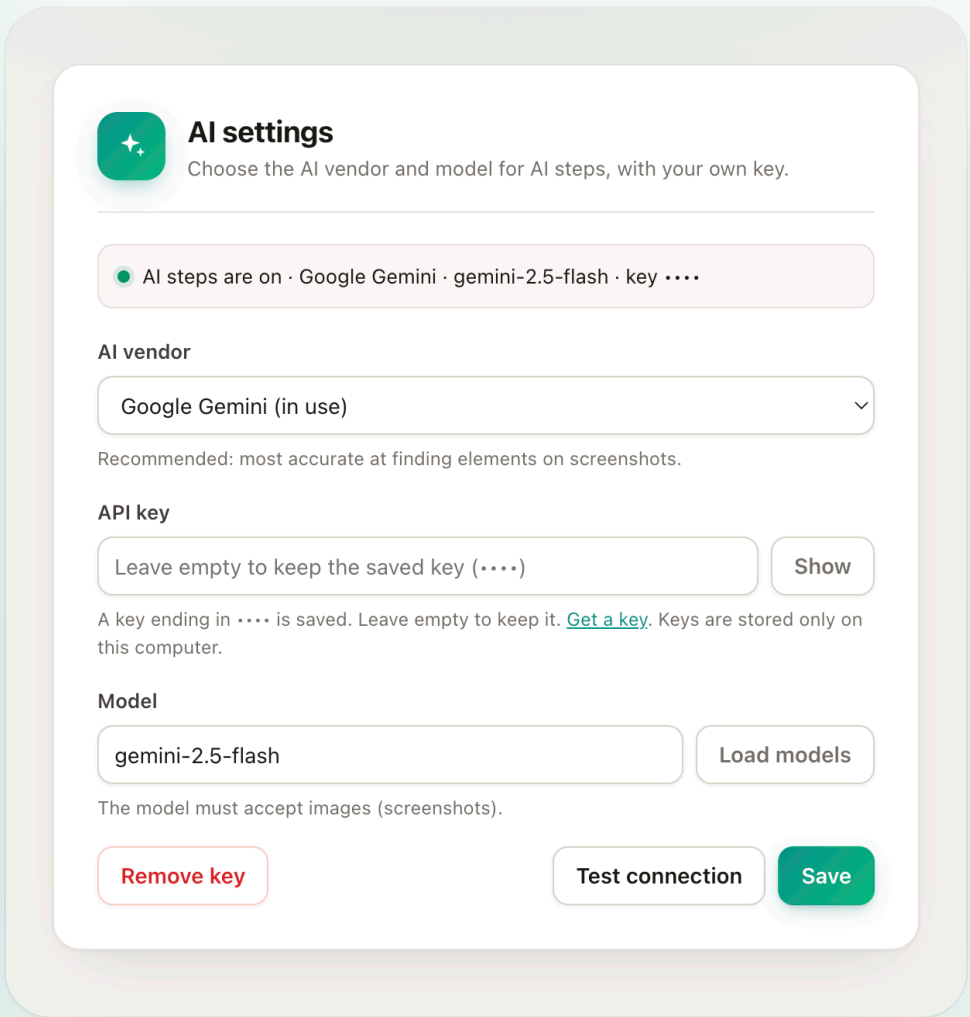


AI THAT STAYS EDITABLE

QA Pilot designs tests, fixes flaky ones, explains failures.

Generate test cases from a feature, test an app on a real phone, write automation code or analyse a failed CI build. Bring your own AI model and key.

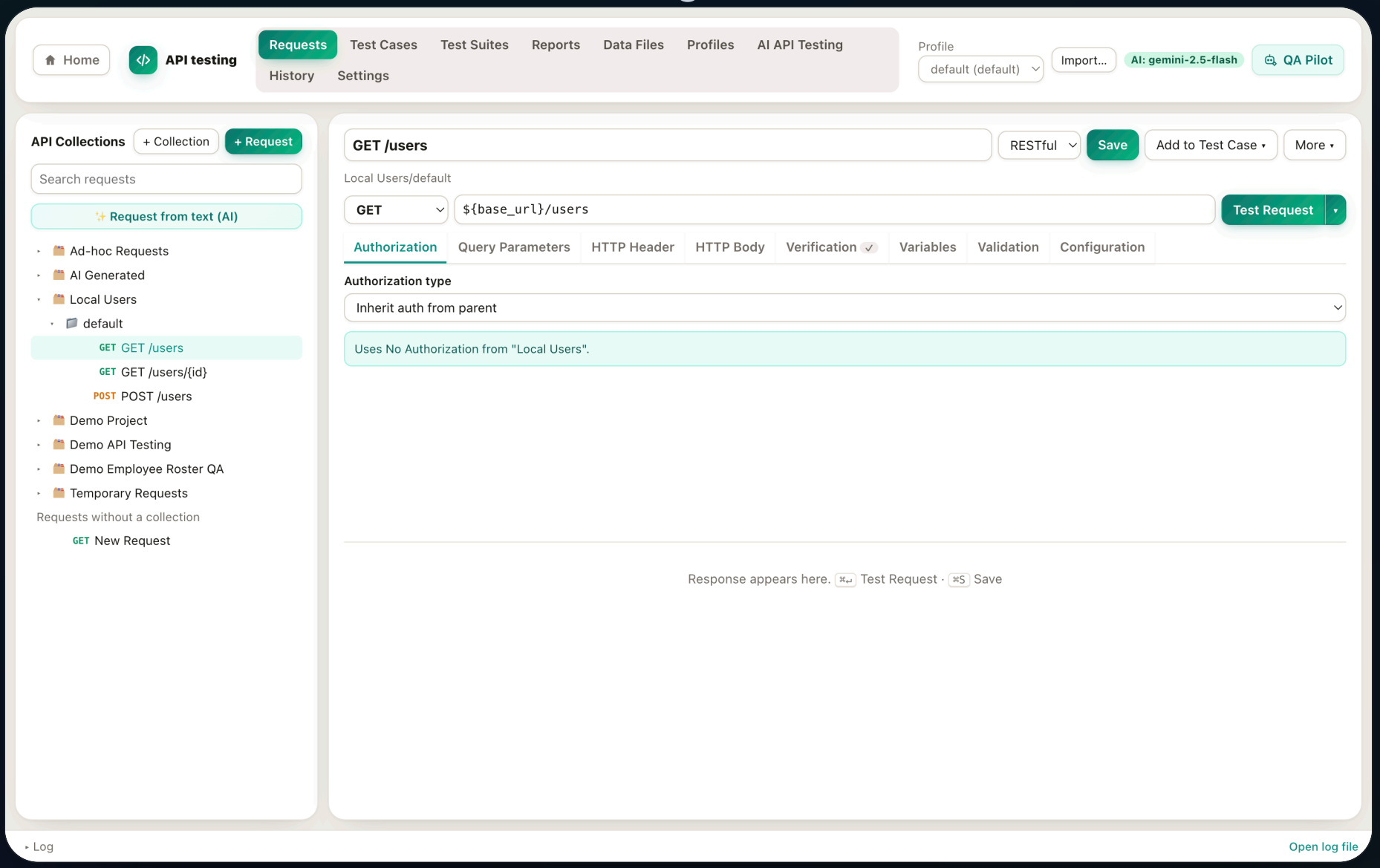


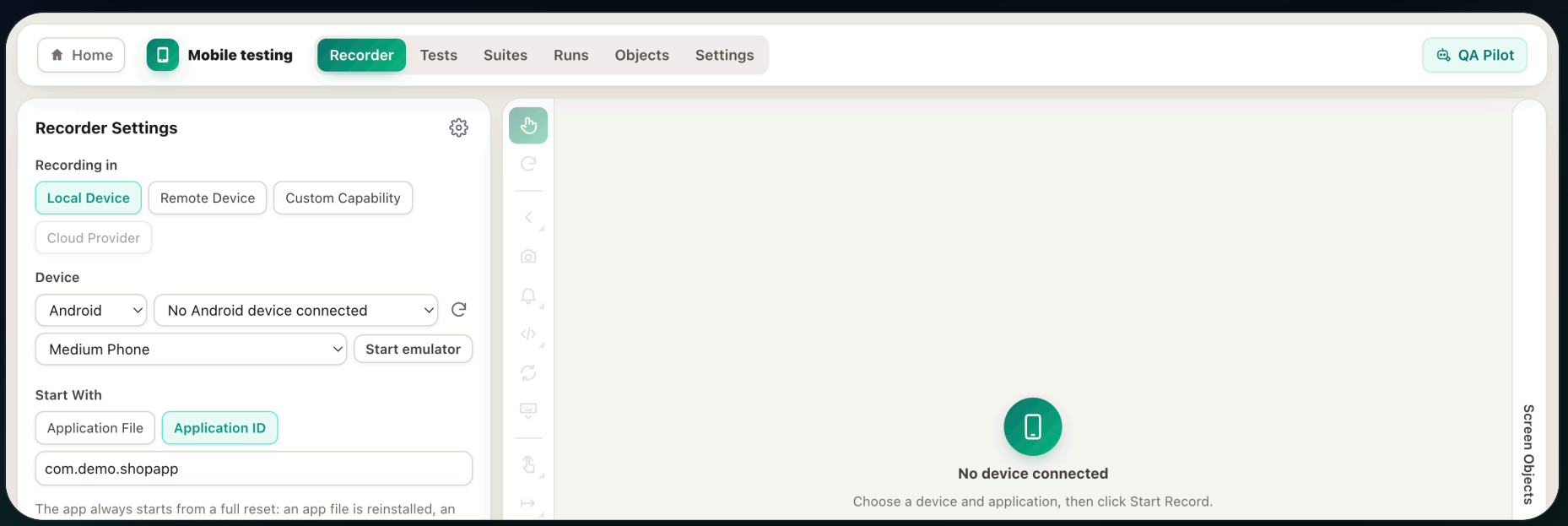


ONE JOURNEY, ONE TOOL

Web, mobile and API under one roof.

REST, SOAP and GraphQL requests with verification and schema checks. Android and iOS recording on local or remote devices.

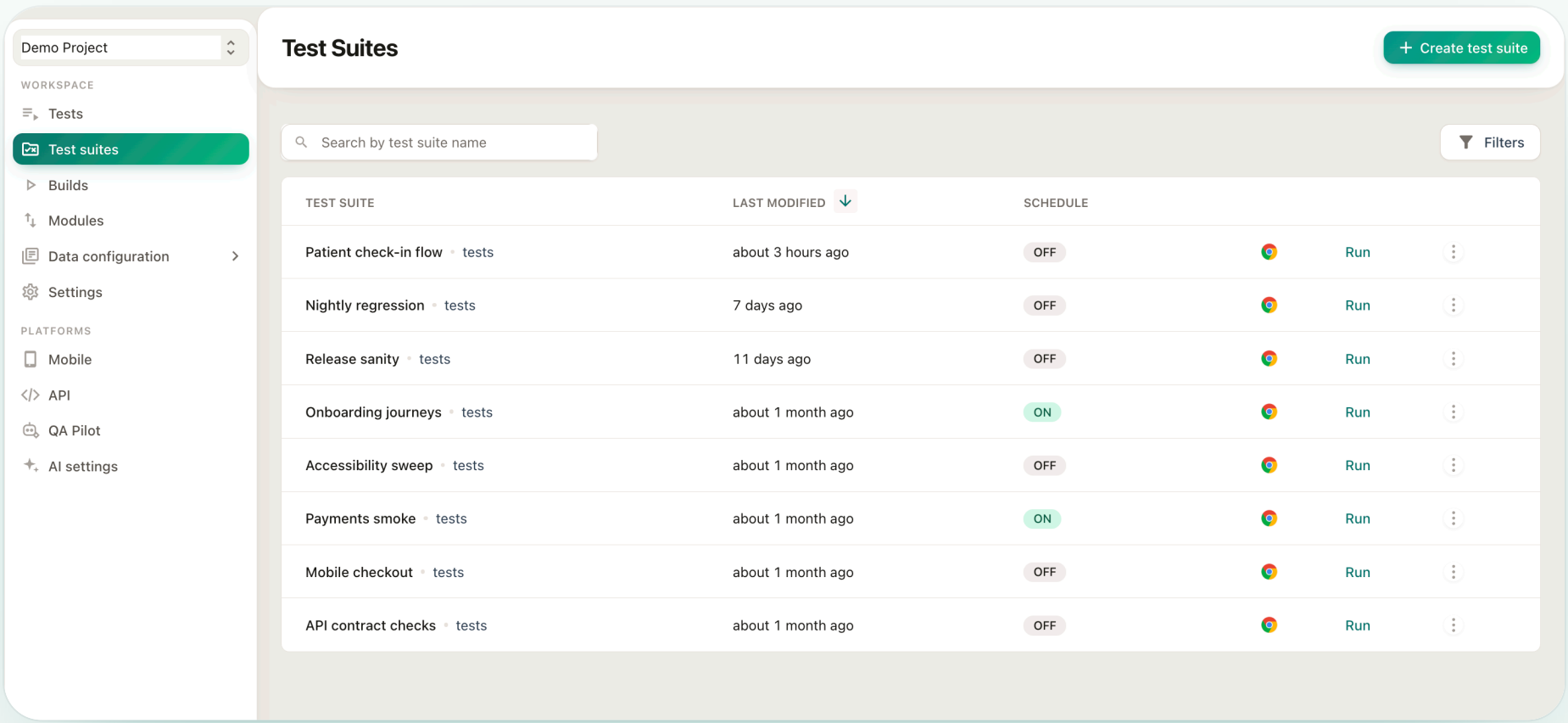


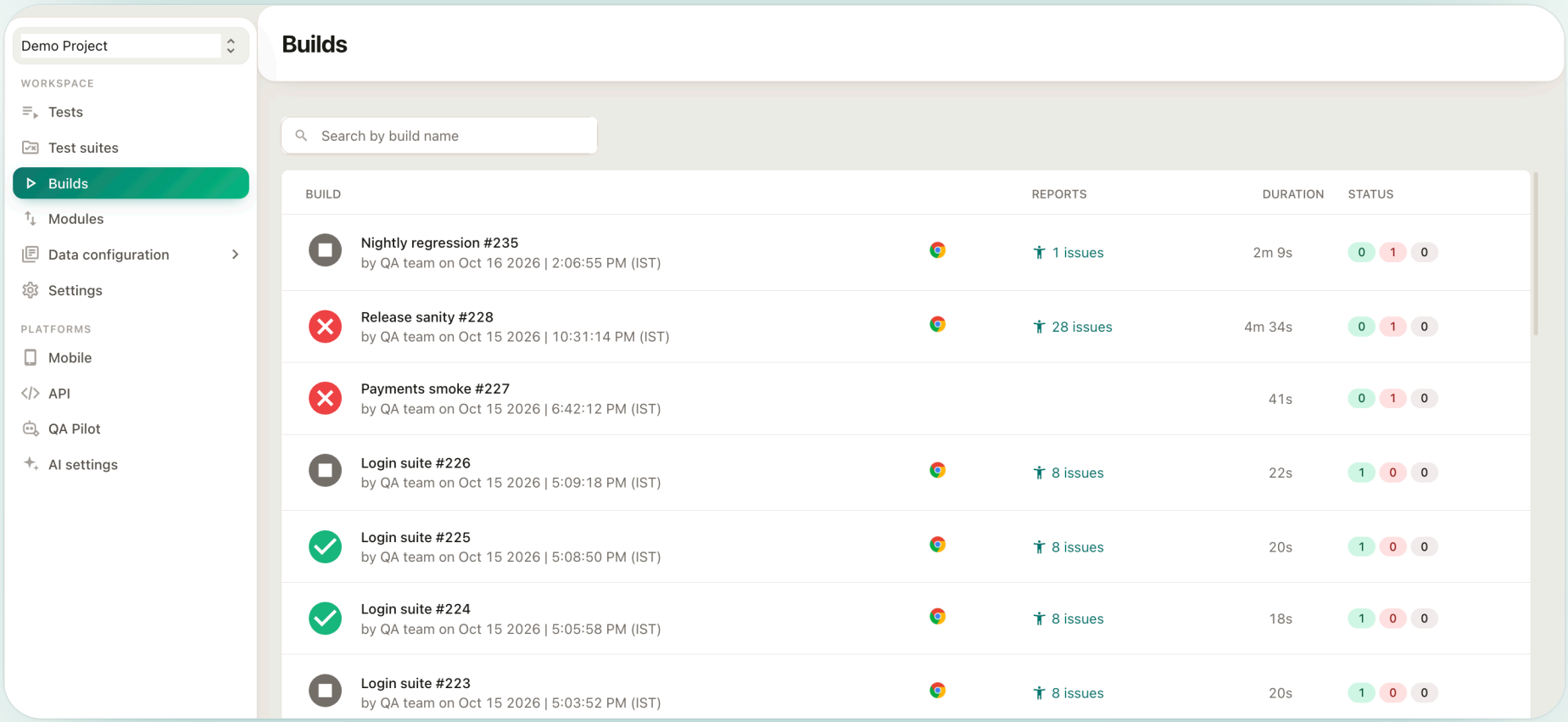


ORGANISE AND SCHEDULE

Suites, schedules and a full build history.

Group tests into suites, run them on a schedule and see every build with its duration, results and accessibility issues.



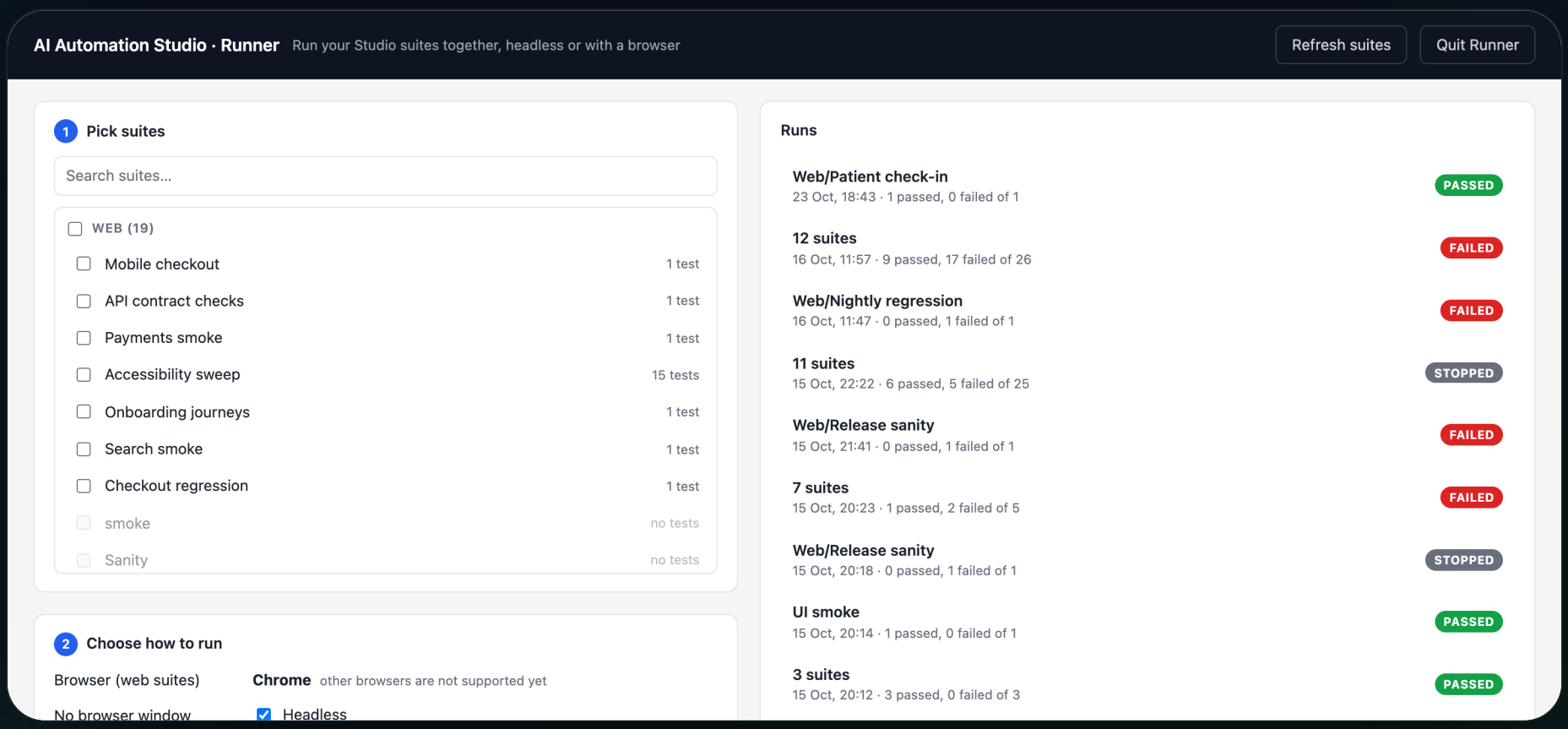


RUNTIME ENGINE

Same tests. Any pipeline.

The aiasc engine runs web, API and mobile suites from the console, in parallel, with retries, profiles, reports and CI-ready exit codes. Or pick suites in the Runner and press Run.

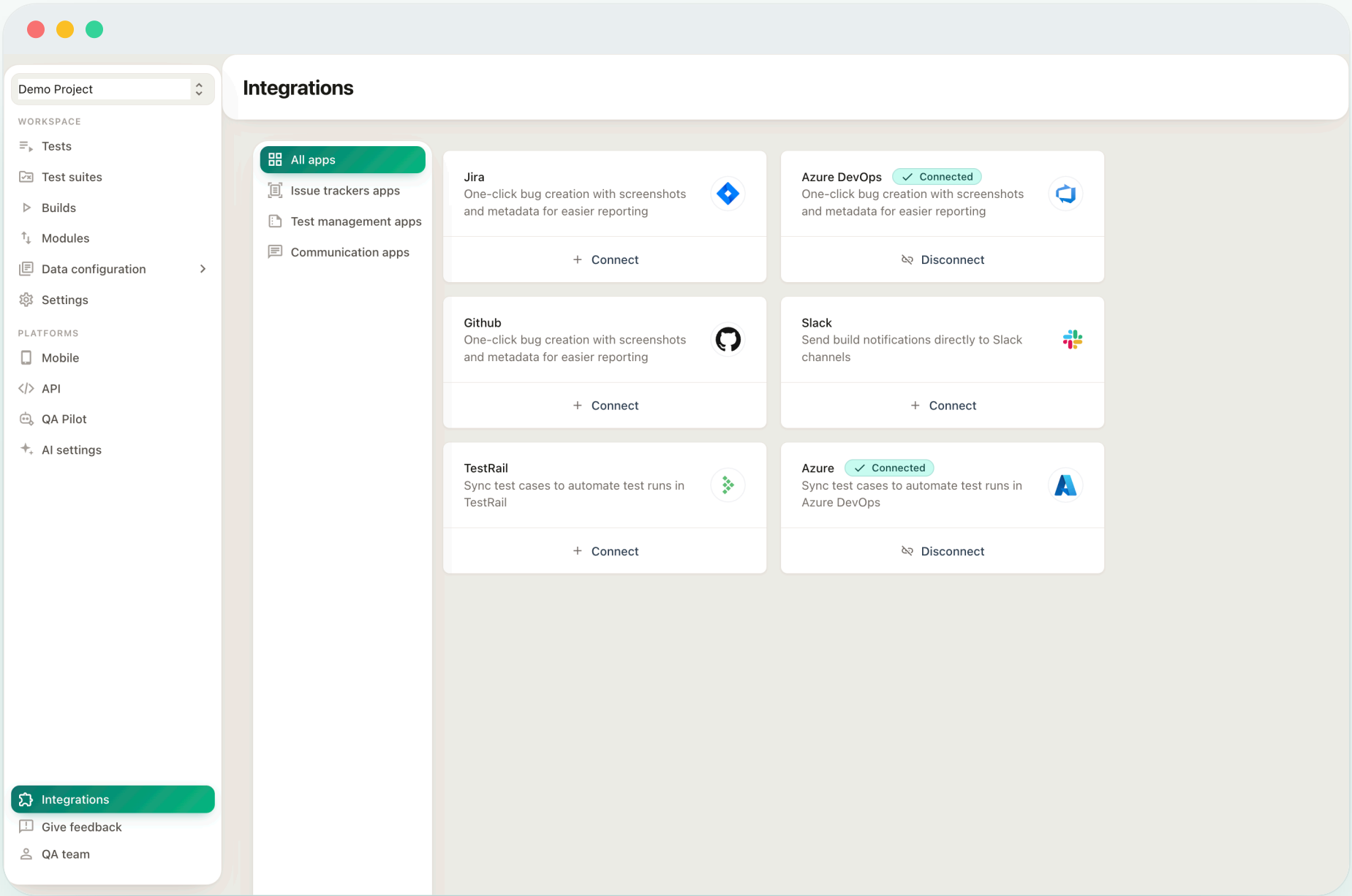
```
$ aiasc -runMode=console -testSuiteCollectionPath="Test
Suites/Collections/Nightly" \
  -executionProfile=staging -retry=1 -retryStrategy=failedExecutions -
report.folderPath=./Reports
INFO aiasc.ConsoleMain - AI Automation Studio Runtime Engine 1.1.0
INFO aiasc.TestSuiteExecutor - START Test Suites/Web/Login suite · Chrome
(headless)
INFO aiasc.TestSuiteExecutor - Test Suites/API/Employee API contract suite
- PASSED
INFO aiasc.TestSuiteExecutor - Test Suites/Mobile/Checkout - PASSED
Total: 26 | Passed: 25 | Failed: 1 | Error: 0 | Time: 4m 12s · Exit code: 1
```



RESULTS WHERE TEAMS WORK

Azure DevOps, Jira, Slack, GitHub and TestRail.

Sync test cases to test plans, raise bugs with screenshots and send build results to the channel your team already reads. Email reports included.



WHY IT IS DIFFERENT

Built for trust, not just for demos.

Local-first	Tests, screenshots and history stay on your machine. No remote session is needed to record or run.
One test, every layer	UI, API, mobile, data, email and accessibility are steps in the same flow.
AI you can edit	Every generated step is a normal step you can review, change and replay.
Honest failures	A step that did not really happen fails the run, instead of passing silently.
Studio to CI	The Runtime Engine runs the same suites in pipelines, Docker or a parallel Runner.



Tests you can trust.

Record it. Run it anywhere. Know when it really failed.

3

platforms in one test: web, mobile and API

10

building blocks beyond clicks and typing

1

engine from the desktop to CI

Comment "DEMO" for a live walkthrough

AI Automation Studio