Make Coding Agents Test and Repair Their Own Work

Coding Agents Alpha Tracker

2026-07-13

Make Coding Agents Test and Repair Their Own Work

By Coding Agents Alpha Tracker • July 13, 2026

Today's strongest practitioner signal is a closed-loop coding workflow: build from a ticket, test through computer use, repair failures, then review again in a separate thread. Also included: GPT-5.6 task routing, Codex updates, usage-limit changes, and an emerging parallel agent environment.

TOP SIGNAL

Make the agent prove the feature works before it reports back. Ross Mike's Codex workflow is a closed loop: pull work from Linear or Notion, build it, run end-to-end browser QA through computer use, repair failures, and return only after the test passes. A separate self-review thread and scheduled PR checks extend that same feedback-loop pattern beyond a single task. [1]

TRY THIS

- **Turn a feature ticket into a build–test–repair loop.** Connect the agent to a Linear or Notion issue (or attach it through MCP), then use this instruction:

  > Build this feature. **Success criteria:** use computer use to test it end to end. If it does not succeed, fix it and repeat. When it's done, report back.

  Ross Mike uses this especially for UI work, where code alone is not sufficient evidence that a form or flow works. [1]

- **Force a clean-context self-review before merge.** Open a *new* thread and ask Sol to review the code it changed, assign a 1–5 score, then tell it:

"Keep fixing until you give yourself 5/5." Ross Mike reports that the agent can be harsh on its own work in this setup. [1]

- **Schedule the boring review work.** In Codex's **Schedules** tab, set a daily job such as: "At 8am, review all open PRs, identify which need further review, run a security review, and spin up a thread to fix anything necessary." The result is a report plus targeted repair threads rather than a manual review queue. [1]
- **Route by task—and constrain the executor.** Theo's current default is **Sol on High**; he recommends it for uncertain work or tasks expected to run longer than 10 minutes. Use **Terra** for budget-sensitive review or implementation work, and reserve **Luna** for model-orchestrated bulk tasks. For Sol, provide sources, examples, style guides, and a clear outcome, then review and redirect: Theo warns it can turn a five-line change into a large rewrite without guardrails. [2]

WHAT SHIPPED

- **GPT-5.6 family:** Sol is the flagship, Terra the balanced everyday model, and Luna the cost-efficient tier; the family is generally available after limited preview. Programmatic Tool Calling is described as letting the model write and run lightweight programs to coordinate tools, process intermediate results, and adapt its next action. [2]
- **Codex developer upgrades:** direct editing of Markdown and code, in-app PR review, Pro mode in Chats with handoff to Codex, Ultra mode, and new documentation. [3]
- **Sol usage tuning:** Codex reverted Sol's product context limit from **372k to 272k** after higher-than-intended usage consumption, with a plan to restore 372k after tuning. The team says inference optimizations should yield roughly **10% more usage**, is fixing excess multi-agent use in high/xhigh reasoning, and has temporarily removed the five-hour limit. [4]
- **Orca:** Jason Zhou says that after trying "almost every parallel ADE," Orca is his current favorite. Its listed features include a native TUI and file viewer, custom commands, mobile support, Claude Code/Codex usage tracking, design mode, and GitHub-to-agent task tracking. Repo: stablyai/orca. [5]

GO DEEPER

- **10:39–12:12 — Computer-use QA loop.** Watch the exact feature-ticket-to-browser-test workflow, including why the agent is told to repair failures rather than merely report them.

OpenAI Just Merged ChatGPT and Codex. This Changes Everything. (10:38)

- **16:32–17:31 — Self-review until 5/5.** A short walkthrough of splitting review into a fresh thread, having the author-agent score its own changes, and iterating on the findings.

OpenAI Just Merged ChatGPT and Codex. This Changes Everything. (16:32)

- **30:23–32:14 — Model routing inside the GPT-5.6 family.** Theo's practical breakdown of when to use Sol, Terra, and Luna—useful if cost and long-running reliability matter more than chasing a single default model.

GPT-5.6: The Review (31:21)

Editorial take: the useful agent loop is increasingly simple—give it a concrete goal, a feedback mechanism, and permission to iterate; then keep cost and scope under deliberate control. [1, 4]

Sources

1. OpenAI Just Merged ChatGPT and Codex. This Changes Everything.
2. GPT-5.6: The Review
3. X post by @ajambrosino
4. X post by @thsottiaux
5. X post by @jasonzhou1993