
ELLM

How to Run a Specialist Agent

Learn how to effectively run specialist agents for tasks like testing, documentation, and bug-fixing with eLLM Code. The article guides you through setting up and using built-in and custom agents, ensuring efficient task management.

AUTHOR

Paul Flanders

PUBLISHED

10 Sep 2026

DOCUMENT

InfoHub — Knowledge, news, and updates from
EssingtonITS

How to Run a Specialist Agent

Paul Flanders · 10 Sep 2026

Estimated time: 4 minutes

Difficulty: Intermediate

Why you'd use this

Some jobs are better done by a focused role with its own instructions than by the general assistant: writing and running tests, bringing documentation up to date, or chasing one bug to its root cause. eLLM Code ships three such agents and lets you add your own as a markdown file. You hand an agent a task, it works through the normal read, edit and run loop, and it reports when it is finished, clearly labelled if anything was left unverified.

Before you start

Permissions required:

- Write access to the workspace. Agents propose edits and run commands like the main assistant, subject to the same Apply and Run approvals and toggles.

You'll need:

- A one-line task for the agent, or nothing at all to use its default brief.
- For a custom agent: a file at `.project-ai/agents/<id>.md` whose first heading is the agent's name and whose body is its role prompt.

Steps

1. Press / and choose **Run specialist agent (QA / Docs / Bug-fix)**, or run **eLLM: Run Specialist Agent**.

2. Pick an agent from the list: **QA Agent** writes and runs tests and reports coverage gaps; **Documentation Agent** updates README, docs and doc-comments and keeps architecture and devlog current; **Bug-fixing Agent** reproduces, diagnoses, fixes and verifies a bug. Custom agents appear below the built-ins.
3. Type the task in the input box, for example "verify the checkout flow", or press **Enter** on an empty box for the agent's default task.
4. An "Agent" activity line names the agent. Review its cards and commands as usual.
5. When the agent finishes, a note reads "finished" or "finished, unverified" with the reason. If it finished unverified, read the warning above it before relying on the result.
6. To chain the QA and Documentation agents automatically after every Plan Mode build, open ⚙️ and tick **Hand off to QA + Docs agents on build completion**.

What you should see

The QA agent adds test files where the project keeps them and runs them. The Documentation agent proposes changes to README and doc files. The Bug-fixing agent is also started automatically when a Plan Mode build finishes with unresolved preview errors, carrying those errors with it.

Troubleshooting

- **A custom agent does not appear:** the file must be inside the memory folder's `agents/` subfolder with a `.md` extension, and its id (the file name) must not clash with `qa`, `docs` or `bugfix`.
- **The agent finished after a malformed action:** it is given two repair attempts, then finishes labelled unverified. Run it again with a narrower task.

Related guides

- How to Apply Skills to a Session
- How to Understand the Verification Gates