
ELLM

How to Plan and Build a Feature with Plan Mode

Learn how to use Plan Mode to organise and build a software feature, from planning requirements to verifying the final build. Choose your level of autonomy and follow step-by-step guidance to ensure a smooth development process.

AUTHOR

Paul Flanders

PUBLISHED

10 Sep 2026

DOCUMENT

InfoHub — Knowledge, news, and updates from
EssingtonITS

How to Plan and Build a Feature with Plan Mode

Paul Flanders · 10 Sep 2026

Estimated time: 8 minutes

Difficulty: Intermediate

Why you'd use this

A one-line request is fine for a small edit, but a whole feature needs requirements, a design and an ordered list of tasks before code is written. Plan Mode runs that sequence for you: it asks clarifying questions, writes the plan into project memory, then works through a live task checklist, verifying the result before it declares the build complete. You choose how much autonomy it gets, from planning only to a hands-off build.

Before you start

Permissions required:

- Write access to the workspace, including the `.project-ai/` memory folder which Plan Mode creates if missing.

You'll need:

- A description of the feature: what it should do, any constraints, and where it lives in the project.
- A decision on the level: **Guided** pauses after each planning stage for your review, **Autopilot** plans and builds through to completion, **Assist** plans only and changes no code.

- For a smooth Autopilot run, the **Auto-apply changes** toggle on; commands run automatically during the build stage in either Guided or Autopilot unless they are dangerous.

Steps

1. Press / in the composer and choose **Plan & build (plan mode)**, or run **eLLM: Plan & Build (Plan Mode)** from the Command Palette.
2. Pick the level: **Guided**, **Autopilot** or **Assist**. A plan banner appears at the top of the panel showing the level, with an **Exit** link.
3. Type the feature description in the composer and press **Enter**.
4. If the planner needs more information, a question form appears inside the chat. Answer each question in its box, pick an offered option, or type in the free-text field, then submit. Leave a question blank to let the planner choose.
5. Watch the planning stages: requirements, then architecture, then a task breakdown. In Guided mode a **Continue** bar appears after each stage; read the stage output, then click **Continue** or type corrections.
6. When the task list appears, the task panel above the composer shows the count and the current task. Click its header to expand or collapse the full checklist.
7. During the build, review and apply diff cards as usual, or leave Auto-apply on. Ticks appear on tasks as they finish.
8. When every task is ticked, the verification gates run: a static import check, the project's own test or build command, the formatter and linter, and for a web app a headless browser load. Failures are fed back and fixed before completion.
9. A self-review pass then checks the result against the plan. On success a completion note appears and the checklist closes.

What you should see

The plan itself is written to `.project-ai/` : requirements and decisions in memory, the design in `architecture.md` , the checklist in `tasks.md` and a summary in `devlog.md` . If the planner recommends skills for the task, a suggestion box appears with an **Enable selected** button. The completion note reports what was verified: tests passed, static check clean, preview clean.

Troubleshooting

- **The run pauses with a reason:** see the guide on stopping, pausing and continuing. Continue sends a targeted nudge; typing guidance steers it.
- **"Several planning steps produced no task list":** the model is not producing a checklist. Type the task list yourself in plain bullets and it is adopted.
- **The build completes but a task was not really done:** nothing is ticked without a file being saved or a command being run that turn, and the final note names the evidence. If something is still missing, describe it and the build continues.
- **You want to leave the plan and chat normally:** click **Exit** on the banner. The memory files stay.

Related guides

- How to Generate a Specification with the Guided Intake Form
- How to Resume a Project and Catch Up
- How to Understand the Verification Gates
- How to Stop, Pause and Continue a Run