

Workflow with AI agents

2025

The Goal: Cognitive Offloading

The primary benefit of an AI agent like Copilot is its ability to handle repetitive, boilerplate, or syntactically complex tasks. This frees up your mental energy to focus on what truly matters: application logic, user experience, and robust architecture. By mastering this workflow, you spend less time on the "how" and more time on the "what" and "why."

Mongo Compass analogy

The screenshot shows the Mongo Compass search interface. The search bar contains the text: "banknotes that have non-english characters in it's letterings". Below the search bar, the query is displayed as: ": { \"\$regex": "[^\u00-\u7F]" }, { \"reverse_lettering\": { \"\$regex": "[^\u00-\u7F]" } }] }". The interface includes standard search controls: Explain, Reset, Find, and Options.

Comparison

```
{ "$or": [ { "obverse_lettering": { "$regex": "[^|\x00-\x7F]" } },  
          { "reverse_lettering": { "$regex": "[^|\x00-\x7F]" } } ] }
```

banknotes that have non-english characters
in their letterings

This is just plain english

~~Code completion~~

SWE-Agents

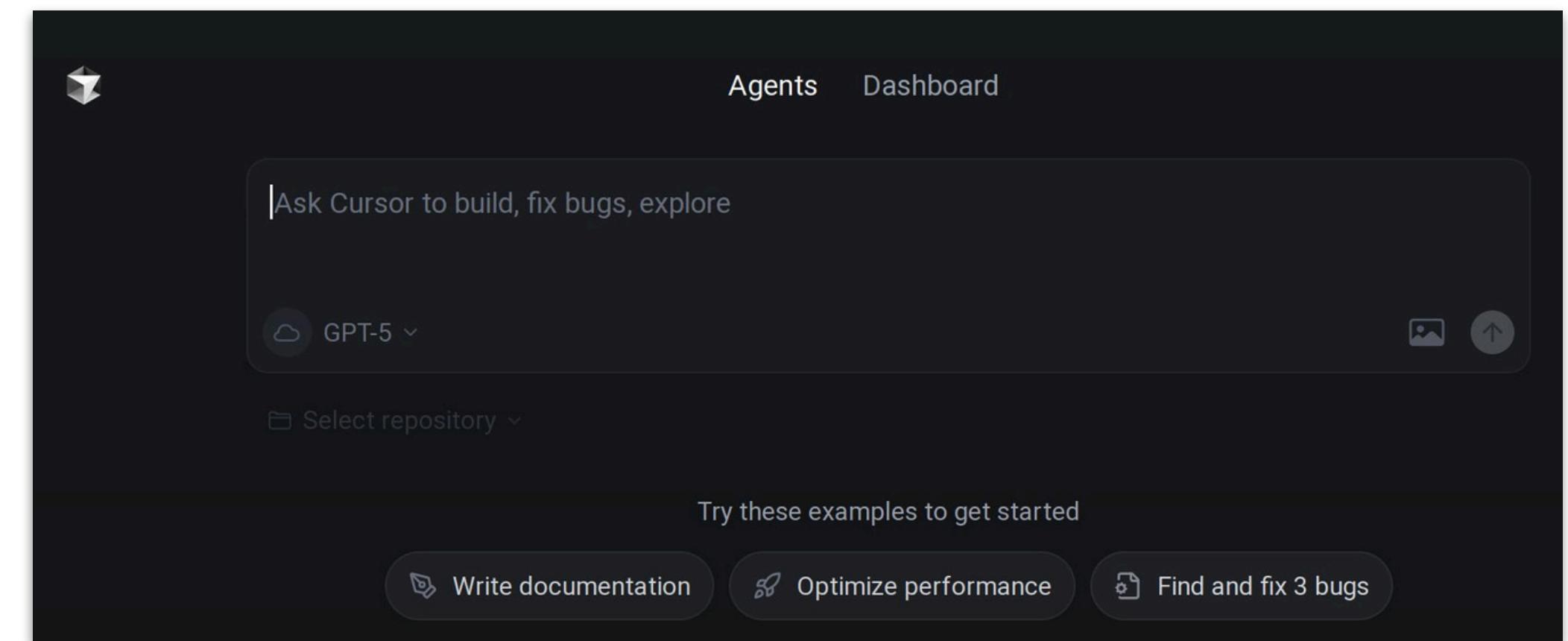
New features

Bug fixing

Code optimization

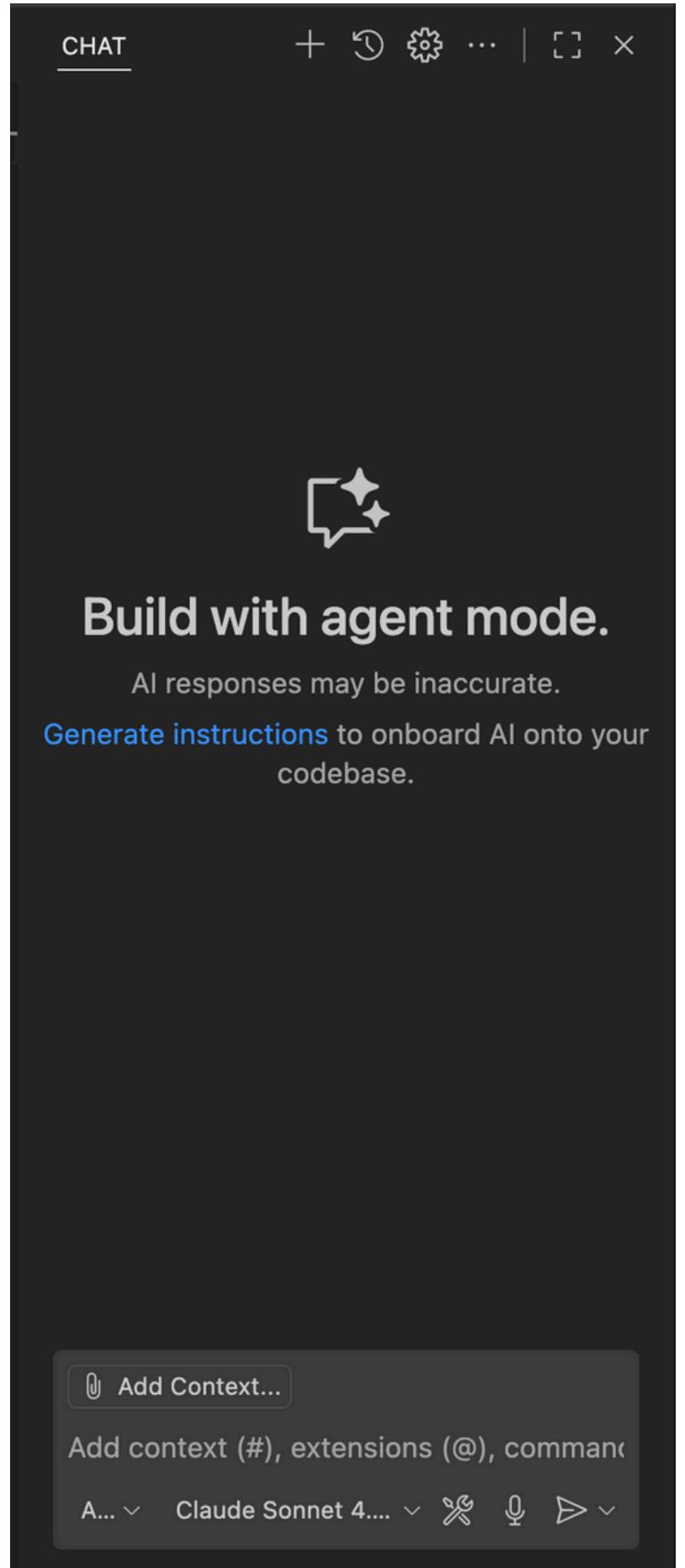
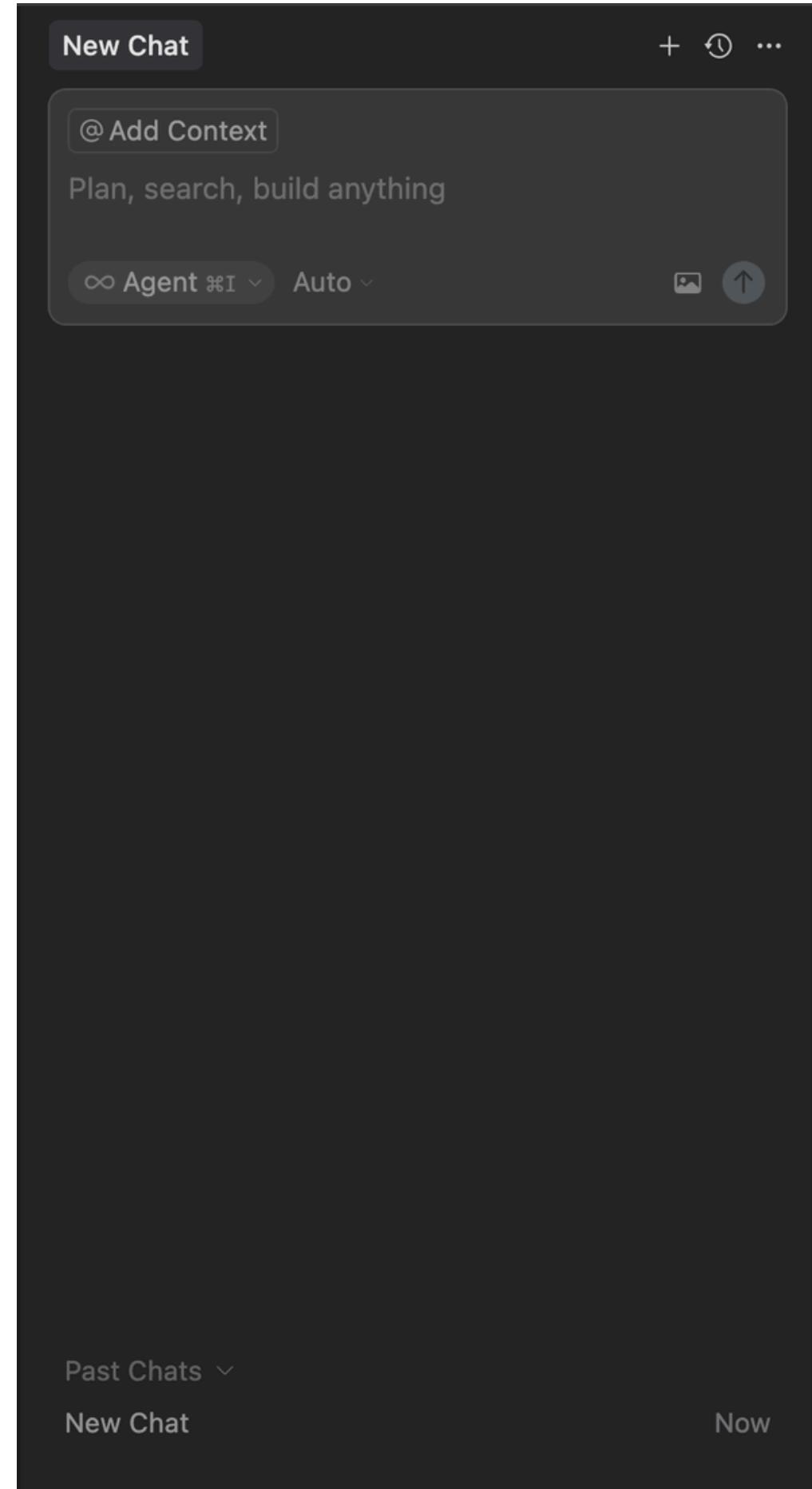
Documentation

...



Cursor/Copilot

Free/Educational/Enterprise



Steps	Description	Example
Provide context	Use dedicated tool or prompt to outline the repo and generate .md file, find potential examples of solving your task	<i>Using “Generate Instructions” in copilot and finding examples of bentoml to triton refactoring</i>
Plan the task	Use dedicated tool or prompt to divide your task into subtasks and generate a tasks.md file	<i>Using prompt or “task master” to generate subtasks into tasks.md</i>
Prompt and iterate	Prompt the steps one by one while being clear and resetting the conversation if needed or adding additional info	<i>Using prompts like: “Okay, now let’s get to the step 1 of #tasks.md. Do ...”, “Okay, but let’s also add ...”</i>
Review	Review the job done, make sure everything is clear and clean. Generate docs or save prompts for the future	<i>Using prompt like: “Now update README.md”, “What can be refactored to remove redundancy?”</i>

Demo

copilot-instructions.md or agents.md

example/workspaces

tasks.md

general prompts

.prompts.md

MCP

Copilot Review

Beast Mode

Links

1. <https://www.youtube.com/watch?v=1L509JK8p1I&t=82s> (Task master with claude)
2. https://www.youtube.com/watch?v=dutyOc_cAEU&t=797s (VSC + postgres MCP)
3. <https://www.youtube.com/watch?v=CxbHw93oWP0> (Prompt engineering)
4. <https://www.youtube.com/watch?v=Kl8XPCZ6CHM> (Copilot vs Cursor)
5. <https://github.com/github/awesome-copilot/blob/main/chatmodes/4.1-Beast.chatmode.md> (Beast mode and other enhancements)