

Coding Nexus

Member-only story

The AI Prompt That Makes You Code Like a 10x Engineer.



Code Coup

Follow

4 min read · 3 days ago

43

3



...

Every now and then, you meet someone at work who *flies*. Like, while the rest of us are still figuring out where to start, they're already pushing commits and reviewing pull requests.

There's one engineer on our team — easily five times faster than anyone I've worked with before. I don't just mean typing speed. He somehow progresses through projects as if he already has the map in his head.

Turns out, he does.

He's created this incredibly effective AI prompt that helps him plan engineering projects step by step — like a mini project manager who actually knows how to code.

When I asked him to share it, and after trying it myself, I understand why. It combines structure, some coding intuition, and a whole lot of clever prompting.

Let me explain how it works.

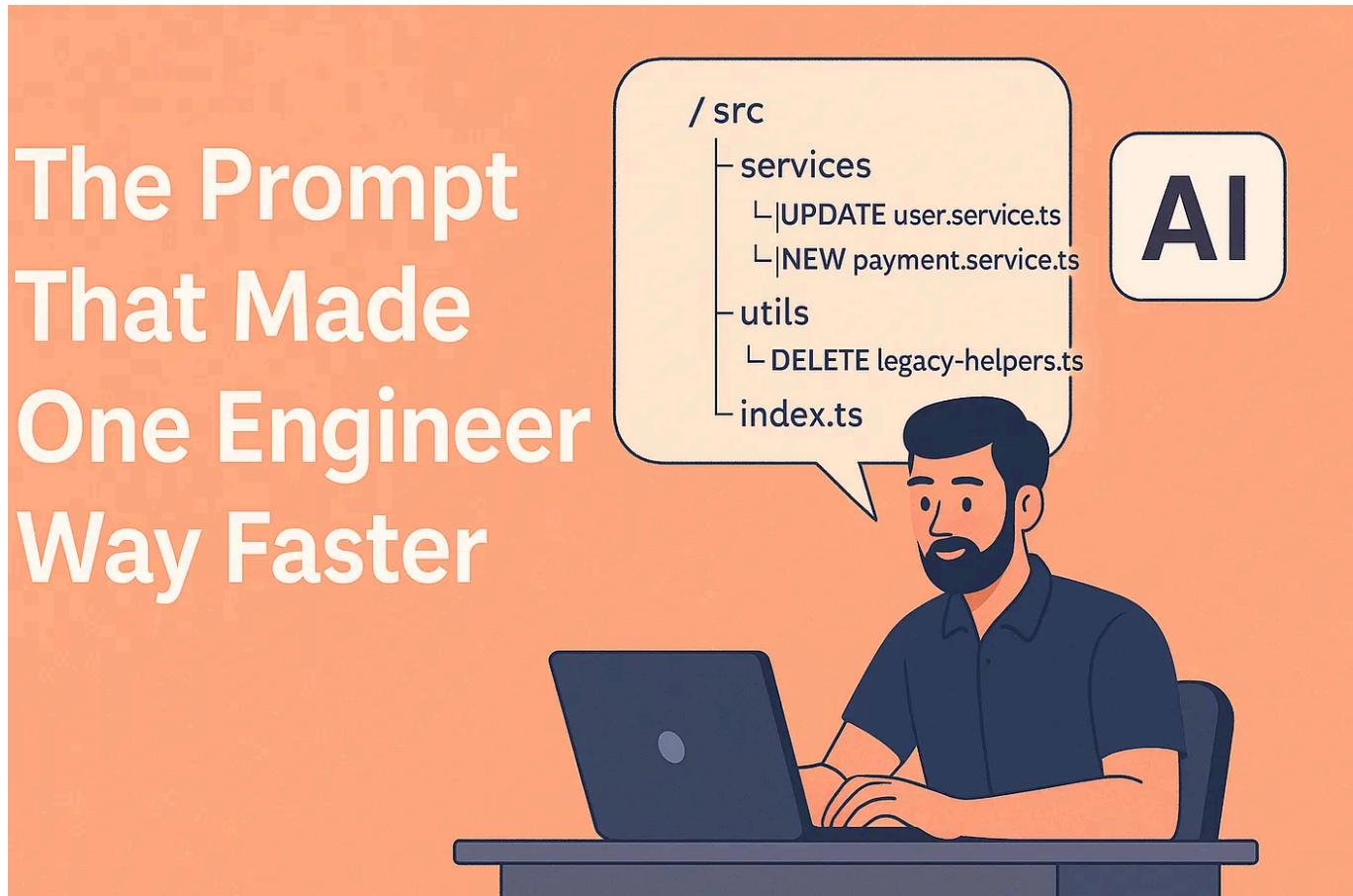


Image is generated by AI — ChatGPT

Step 1: Allow the AI to Examine the Codebase

The first thing this prompt does is... well, *look around*.

It asks the AI to explore the project — list out the folders, peek into `docs/`, and skim through files like `AGENTS.md` or `CLAUDE.md`.

It's like saying, "Don't just jump in — get your bearings first."

Here's an example of what it might produce:

```
/src
  └── api
  └── services
  └── utils
  └── tests
```

Now you can instantly see what's where. No more searching for ten minutes wondering where the helpers are.

Step 2: Ask Questions Before Assuming Anything

This part cracked me up the first time I saw it. The prompt literally makes the AI ask clarifying questions before finalising a plan. You know — the kind of questions a thoughtful dev asks in Slack before touching the code.

Things like:

- “Which module should the new helper live in?”
- “Should this endpoint return JSON or HTML?”
- “Are we following REST or GraphQL here?”

It feels surprisingly human. It slows things down just enough to prevent the “oops, wrong folder” kind of mistakes later.

Step 3: Show the File Tree of Changes

Next, the AI generates a mini file map that shows exactly what's going to change. It labels each file so you can see at a glance what's new, what's been updated, and what's being removed.

Example:

```
/src
  └── services
      ├── UPDATE user.service.ts
      └── NEW payment.service.ts
  └── utils
      └── DELETE legacy-helpers.ts
  └── UPDATE index.ts
```

That alone saves a significant amount of mental overhead. You can see the *scope* of the work in one view. No guessing, no “what all did I touch?” panic at review time.

Step 4: File-by-File Breakdown

This is the part where it gets detailed.

The AI goes file by file, describing what'll change — in plain English first,

and then in code. Nothing fancy, just clean and practical.

For example:

File: src/services/user.service.ts (UPDATE)

- Add A `getUserByEmail(email: string)` function that looks up a user by email.
- Refactor `getUserById` So it reuses that logic.

And here's the snippet it might generate:

```
const users = [
  { id: 1, email: "alice@example.com", name: "Alice" },
  { id: 2, email: "bob@example.com", name: "Bob" },
];

export function getUserByEmail(email: string) {
  return users.find(u => u.email === email) || null;
}

export function getUserId(id: number) {
  return users.find(u => u.id === id) || null;
}
```

Get straight to the point. You can include that in your project and proceed.

Step 5: Add Context and Rationale

At the end, it wraps everything up with explanations:

Why the change exists, what might break, and what to test.

Something like:

“The new helper improves lookup flexibility. Add unit tests for both getUserById and getUserByEmail.”

It’s the kind of comment a senior engineer would leave in a PR.

Why It Works

The reason this system works is that it **forces structure before code**.

Most AI prompts go, “Sure, here’s your code.” This one says, “Wait, let’s make a plan first.”

It’s basically teaming up with someone who’s very methodical — except they never tire, and they always keep documentation up to date.

If You Want to Try It

You don’t need to copy the whole thing.

Even borrowing the structure helps:

1. Look through the repo first.
2. Ask clarifying questions.
3. Make a quick file map.
4. Describe changes file by file.

5. Add reasons and test notes.

Try it once. It's oddly satisfying — like tidying your desk before starting a big project. You feel organized, confident, and prepared to code.

Final Thought

This isn't about automating your job. It's about removing the boring parts—so your brain can focus on the creative stuff.

The engineer who created this didn't build a shortcut; he built a *multiplier*.

And that's the whole point of good AI — not to replace you, but to make your best work easier.

AI

Writing Prompts

Prompt

Coding

Ai Agent



Published in Coding Nexus

7.2K followers · Last published 1 hour ago

Follow

Coding Nexus is a community of developers, tech enthusiasts, and aspiring coders. Whether you're exploring the depths of Python, diving into data science, mastering web development, or staying updated on the latest trends in AI, Coding Nexus has something for you.



Written by Code Coup

2.1K followers · 1 following

Follow

Responses (3)



Bgerby

What are your thoughts?



BigBizBot

1 day ago



PROMPT:

Using Dia side chat with the article as context I asked:

Read this article and write the prompt the author is talking about. Include all details.

ANSWER:

The article's prompt is a structured, multi-step request that makes the AI plan changes... [more](#)

1 [Reply](#)



Teo Wave

3 days ago



Excellent

4 [Reply](#)



Syed Munawar Hussain Bukhari

1 day ago



Really enjoyed this hope you'll visit my posts too



[Reply](#)

More from Code Coup and Coding Nexus



 [Code Coup](#)

LangExtract: Google's Game-Changing Python Library for Data...

Ever stared at a mountain of words, a report, a book, maybe a stack of doctor's notes, and...

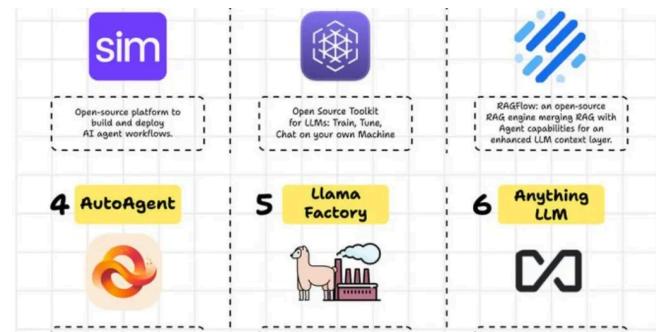
⭐ Aug 13



...

A screenshot of the LangExtract Python library interface. On the left, there is a code editor window displaying a snippet of Python code for an API request. On the right, there is a sidebar titled "Agent memorized" containing three items: "Auth config for MCP setup" (Infrastructure, Jul 22, 2025), "Staging rate limit workaround" (BugFix, Jul 14, 2025), and "Dashboard export logic (v2)" (Frontend, Jul 4, 2025). Below the sidebar, there is a section labeled "Auto" with an upward arrow. At the bottom, there is a toolbar with various icons.

 In Coding Nexus by Algo Insights



 In Coding Nexus by Civil Learning

6 Open-Source AI Projects You Must Try (Agents, RAG & Fine-...

The AI world is complex right now. Every week there's a new repo, a new framework, and a...

⭐ Sep 18



...

A graphic for USAURUS NATIVE LOCAL LLM SERVER. It features a large green cartoon-style T-Rex on the right, holding a small white server tower. To the left of the dinosaur, the text "USAURUS" is written in large, bold, white letters, with "NATIVE LOCAL" in orange and "LLM SERVER" in yellow. The background is dark blue.

 In Coding Nexus by Code Coup

4 Open-Source Tools That Made Me Rethink My Dev Setup

I've been coding for a while now. Most of the tools we use every day... they've been the...

★ Sep 3



...

[See all from Code Coup](#)

[See all from Coding Nexus](#)

Recommended from Medium



 In Data Science Collective by Erdogan T

Build Your Private Language Model: Local and Specialized For...

A complete step-by-step guide from setup to deployment of local language models, makin...

★ 6d ago



...

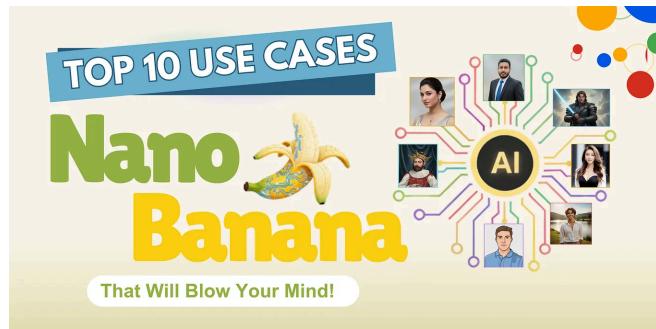
Osaurus: A Native Local LLM Server for Apple Silicon

If you've ever tried running large language models (LLMs) locally on your Mac, you...

★ Sep 7



...



 In Level Up Coding by Yasas Sandeepa

Google's New AI – Nano Banana: Top 10 Use Cases That Will Blow...

How Nano Banana is redefining the future of AI Image Gen

★ 6d ago



...



GLM 4.6



Joe Njenga

I Tested Newly Released GLM 4.6 (And Discovered a Cheaper Way t...)

GLM 4.6 has been released just a few days after Claude 4.5, and it's another coding...

2d ago



...



SONNET
4.5



In Towards AI by Gao Dalie (高達烈)

Why Claude Sonnet 4.5 Is So Much Better Than GPT-5(Codex) And...

The new version of Claude, Sonnet 4.5, has been released. You might be wondering,...

2d ago



...

Welcome to BitNet.
How can I help you today?

What do you want to know?

CPU

By messaging BitNet, you agree to our Terms and Privacy Policy.



In Coding Nexus by Algo Insights

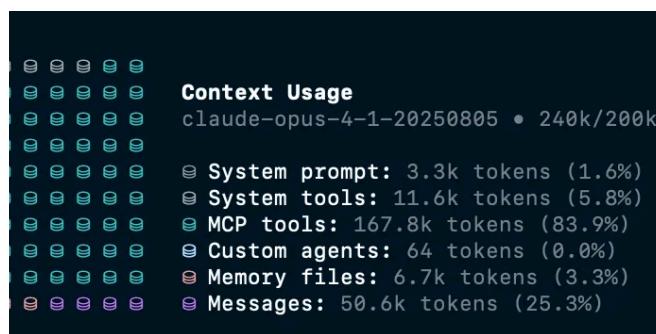
Microsoft Just Declared War on the GPU Mafia: Meet bitnet.cpp

Bitnet.cpp will break the GPU Lock

4d ago



...



Claude Code Limit Hit on Max Plan?! What to do next:

\$200/m plan blocks me for a week!

5d ago 63 15



...

See more recommendations