

Building with Claude Code Subagents (My Beloved Minions)

5 min read · Aug 3, 2025



Yee Fei

Follow



Listen



Share



More

I tried Claude Code's subagents after its recent launch. The feature unlocks more power with Claude Code by allowing me to transition from a general-purpose AI assistant to a specialized “team” of experts. Each expert has a specific role, context, and toolset, which makes my workflows more efficient and reliable — especially for complex tasks like refactoring or debugging.

My Old Workflow: The Generic Task Launcher

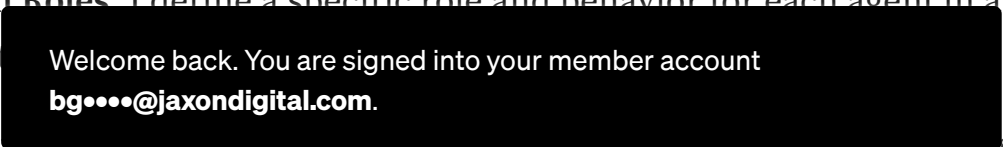
Before subagents, my process was simpler, but less specialized. I'd use the `CLAUDE.md` to set general rules, like an autoscaling policy: “min 2, max 5 agents as applicable”

- It was good for running tasks in parallel, but it was generic.
- There was no specialization. No custom context window for each agent.
- The system prompt was the same for everything. It worked for quick, simple tasks.

What are Claude Code Subagents?

Subagents are specialized AI agents within Claude Code. Instead of one general-purpose agent, now you can define each with their own roles and tools. (Imagine “minions” Bob, Kevin, Stuart etc..).

They're a great way to divide up complex problems into smaller, more manageable tasks, each handled by an expert agent.

- **Custom Roles:** I define a specific role and behavior for each agent in a system prompt. Welcome back. You are signed into your member account bg...@jaxondigital.com.
- **Dedicated Context:** Each agent has its own context window. This keeps it focused and prevents, say, refactoring info from cluttering a testing task.
- **Specialized Tools:** I can give an agent access to only the tools it needs, including any connected MCP server tools. For example, one agent gets access to Supabase MCP tools, but might not need specialised tools used for UI testing.

An Unexpected Benefit: Learning Together

Subagents don't just help me get stuff done, they teach me. Working with them has become a powerful learning tool. As they suggest new frameworks, development tools, or design patterns, I'm constantly expanding my own knowledge. New term or idea that's unheard of? That's a new learning cue!

This new way of working has been a huge boost for my own growth and development — a happy side effect I wasn't expecting (although sometimes it could be intimidating with information-overflow).

To 'summon' a specific minion, you can call its name in the prompt. Its "Description" tells Claude when to call on Bob, Kevin, or Stuart.

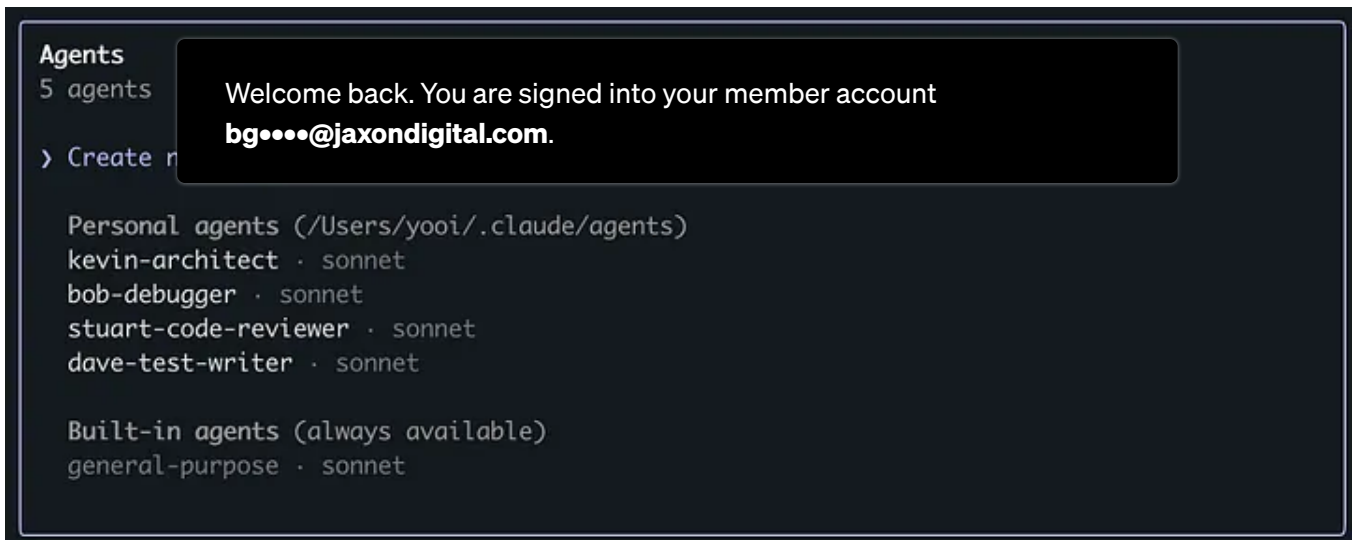
How to Set Up a Subagent

Setting one up is pretty straightforward. You use a built-in command and define its configuration

- Run the `/agents` command in Claude Code.
- Select "Create New Agent."
- Choose between a project-level or user-level agent.
- Define the agent's name, description, system prompt, and tools.

Example Usage: Minions in Action

You can set up agents to handle a development workflow. These minions can function as a team that knows how to hand off tasks to each other. For example:



Kevin the Architect

Kevin is my go-to for big, complex changes. He's an expert at planning and systematic refactoring.

- He plans first, always. My prompt forces Kevin to create a detailed plan and get my approval before he changes a single line of code.
- He thinks critically. He uses Zen MCP tools like planner and challenger to think through every angle.
- He even gets to 'discuss' different perspectives from other models, like Gemini, to avoid rushing into action especially on major changes. (With Zen MCP we can leverage each Model's specific strengths — for example extended thinking with Gemini Pro, fast iteration with Flash etc.). I had fun observing 2 LLMs 'debating' with the 'consensus' tool.

- He knows how to use Context7 MCP to pull in the latest documentation and check for updates.
- Safety is his number one rule. This prompt makes sure he considers security and backward compatibility before starting.

Welcome back. You are signed into your member account
bg....@jaxondigital.com.

So instead of asking a generic Claude Code instance to “refactor this,” I hand the task to Kevin.

When we reach a consensus, we move on to implementation. But if I hit a bug, I have a different minion for that:

Bob the Debugger

Bob is my detective. He’s excellent at finding the root cause of an error

- He digs deep. I give him a bug, sometimes he’ll ask me for more questions and data points to get to the root cause.
- He uses Zen MCP’s debug and tracer tools to analyze the code and the call flow.
- For example, if it looks like a data problem, he’ll use the Supabase MCP tool to check the database schema, config, or policies.
- He proposes a small, targeted fix and even suggests a test to confirm it’s gone for good

That’s when he hands the task over to...

Stuart the Code-Reviewer

Welcome back. You are signed into your member account
bg....@jaxondigital.com.

Stuart is my meticulous code reviewer. He makes sure all my changes are up to the project's standards.

- He's been trained on the project's `CLAUDE.md` file, so he knows all the architectural patterns and best practices.
- He's my second set of eyes. He checks for proper usage depending on the framework and looks for critical security issues that are easy for me to miss.
- He once surprised me by discovering the Zen MCP secaudit tool on his own when I was working on some changes related to auth, which sent us down the entire OWASP rabbit hole lol.

Dave the Test-Writer

Dave is my expert QA automation engineer. His job is to validate that a feature works correctly from the user interface down to the database.

- He designs the test. I tell Dave what I need to test, and he designs a comprehensive test plan that covers all the edge cases.
- He writes the scripts. Dave uses the ZenMaker playwright tool to run tests, and write scripts that simulate real user interactions and the Supabase tool to write read-only queries that validate the database state.
- There are even auto screenshots that save automatically into folders for functional tests.
- He reports the results. After running the tests, he gives me a clear report with pass/fail statuses and specific details on any failures.

You can have more agents based on specialty, for example:

- A security agent that runs secaudit and other security tests.
- A UX/UI agent that specializes in better user experience and design, incorporating accessibility testing.

Beyond these individual agents, I'm also looking into building a more established workflow— For example, after Kevin kick started, I ask the agent to break down to plan work into phases and workstreams. [More on this later and sharing on my good and bad experience and learning]

Video version of what's being discussed in this blog:

Welcome back. You are signed into your member account
bg....@jaxondigital.com.

Claude

Claude Code

Ai Agent

Agentic Ai

Agents



Follow

Written by Yee Fei

33 followers · 5 following

<https://www.linkedin.com/in/yeefei>

No responses yet




Bgerby

What are your thoughts?

Welcome back. You are signed into your member account
bg....@jaxondigital.com.

More from Yee Fei

 Yee Fei


How Claude Code Agents and MCPs Work Better Together 🦊🦊

I've shared how I use sub-agents and parallel workflows with my team of subagent "minions."
Another real breakthrough came when I stopped...

Aug 30  52



Welcome back. You are signed into your member account
bg....@jaxondigital.com.


 Yee Fei

Parallel Development with ClaudeCode and Git Worktrees

My dev workflow with Claude Code has become a lot more flexible. It's no longer about a single agent working on one thing at a time. It's...

Aug 17  53



 Yee Fei

Claude Code Skills—Superpowering Claude Code Agents

Claude Code recently launched Skills, and for weeks, I saw videos and posts about how mind-blowing the feature was. I was eager to try it...

3d ago 🖱️ 2



Welcome back. You are signed into your member account
bg....@jaxondigital.com.



Yee Fei

Automated Screenshot Uploader for Remote Claude (macOS)

This guide provides a robust, step-by-step solution for users running a Claude code environment on a remote server who need to quickly...


Oct 25 🖱️ 51



See all from Yee Fei



Recommended from Medium

Welcome back. You are signed into your member account
bg....@jaxondigital.com.


 Yogeshwar Tanwar

The \$699 Mac Mini Server That Replaced My AWS Bill: The Devil's in the Details (Part 2)

Three months in, and I'm still waiting for the disaster. The catastrophic failure that would send me crawling back to AWS at 3 AM with my...

Oct 13  251  1



 In AI Software Engineer by Joe Njenga


Anthropic Just Solved AI Agent Bloat—150K Tokens Down to 2K (Code Execution With MCP)

Anthropic just announced that they built LLaMA 3.1, which is 98% faster than the previous version, by 98%,
shift from to

Welcome back. You are signed into your member account
bg....@jaxondigital.com.

★ 5d ago



 In Coding Beauty by Tari Ibaba

This insane new coding model is 13 times faster than Claude Sonnet 4.5



Just wow.

★ Nov 2 🤝 248 💬 3





In How To Profit AI by Pump Parade

We Tested

Welcome back. You are signed into your member account
bg....@jaxondigital.com.

Results

Crypto trading is a high-risk activity that can lead to significant financial loss and anxiety when a token drops 20% in minutes.

Nov 3



151



7



Daniel Avila

Claude Code: From Zero to Hero

What is Claude Code?

Jun 22



235



Welcome back. You are signed into your member account
bg....@jaxondigital.com.



The Atomic Architect

You Haven't Seen AI Until You Try Claude Sonnet 4.5's New Feature—It Redefines Insane

I built a working expense tracker in eight minutes while my coffee was still hot, and it remembered every receipt when I closed my laptop...



Oct 26



505



20



See more recommendations