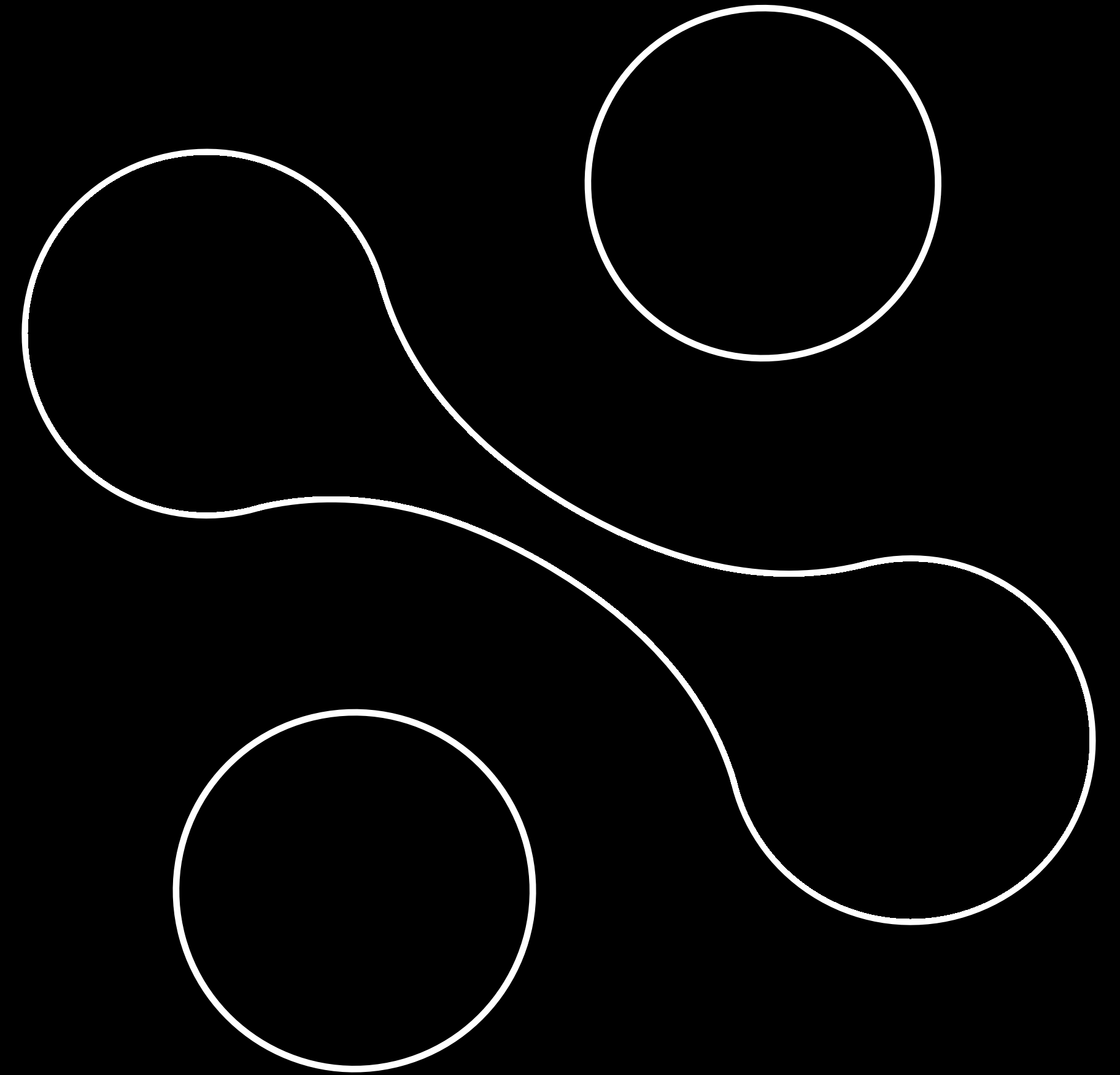
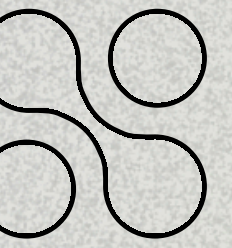


The Self-Improving MCP Server

Continuous Testing and Adaptation Across Clients

Enrico Toniato
CTO Manufact

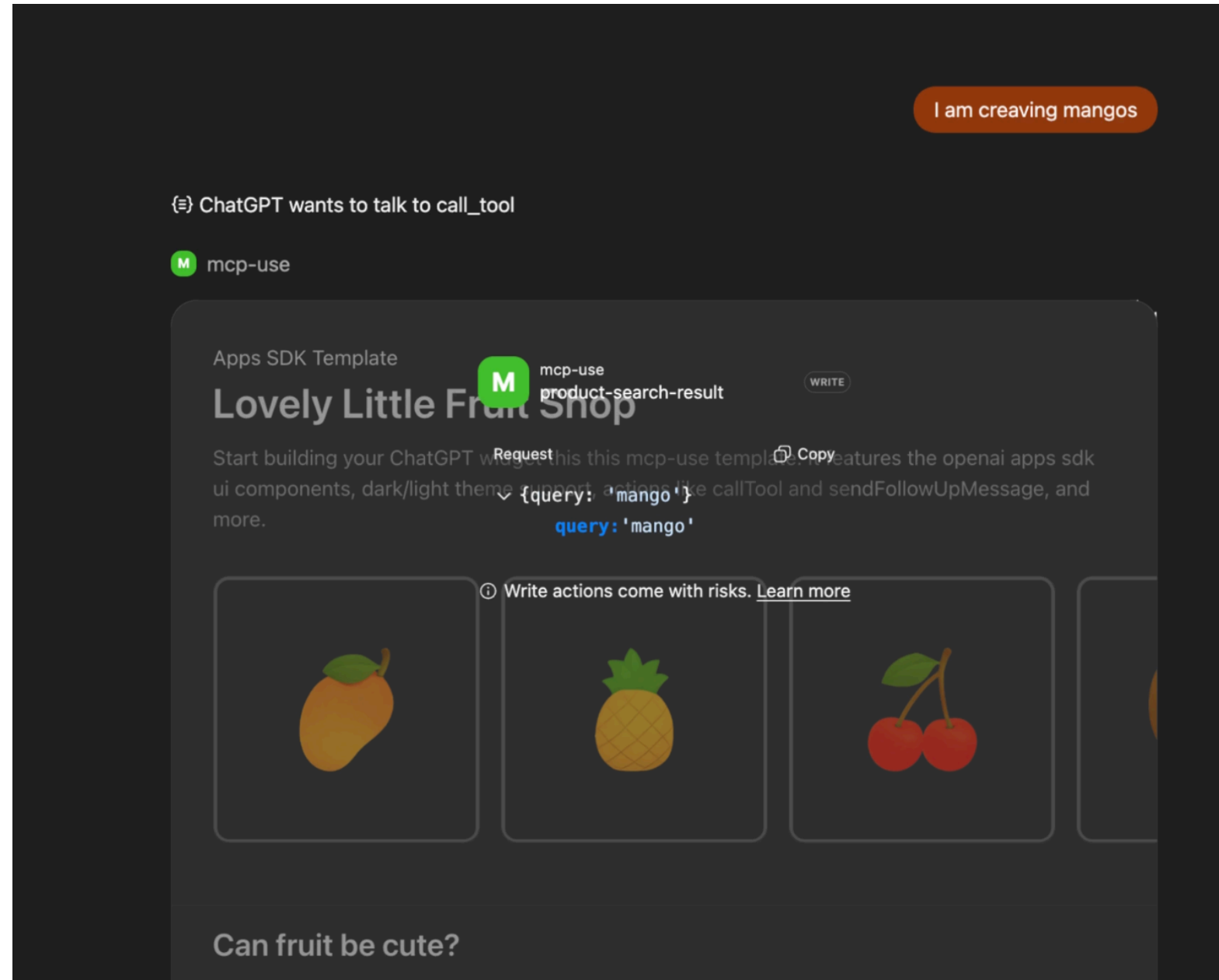




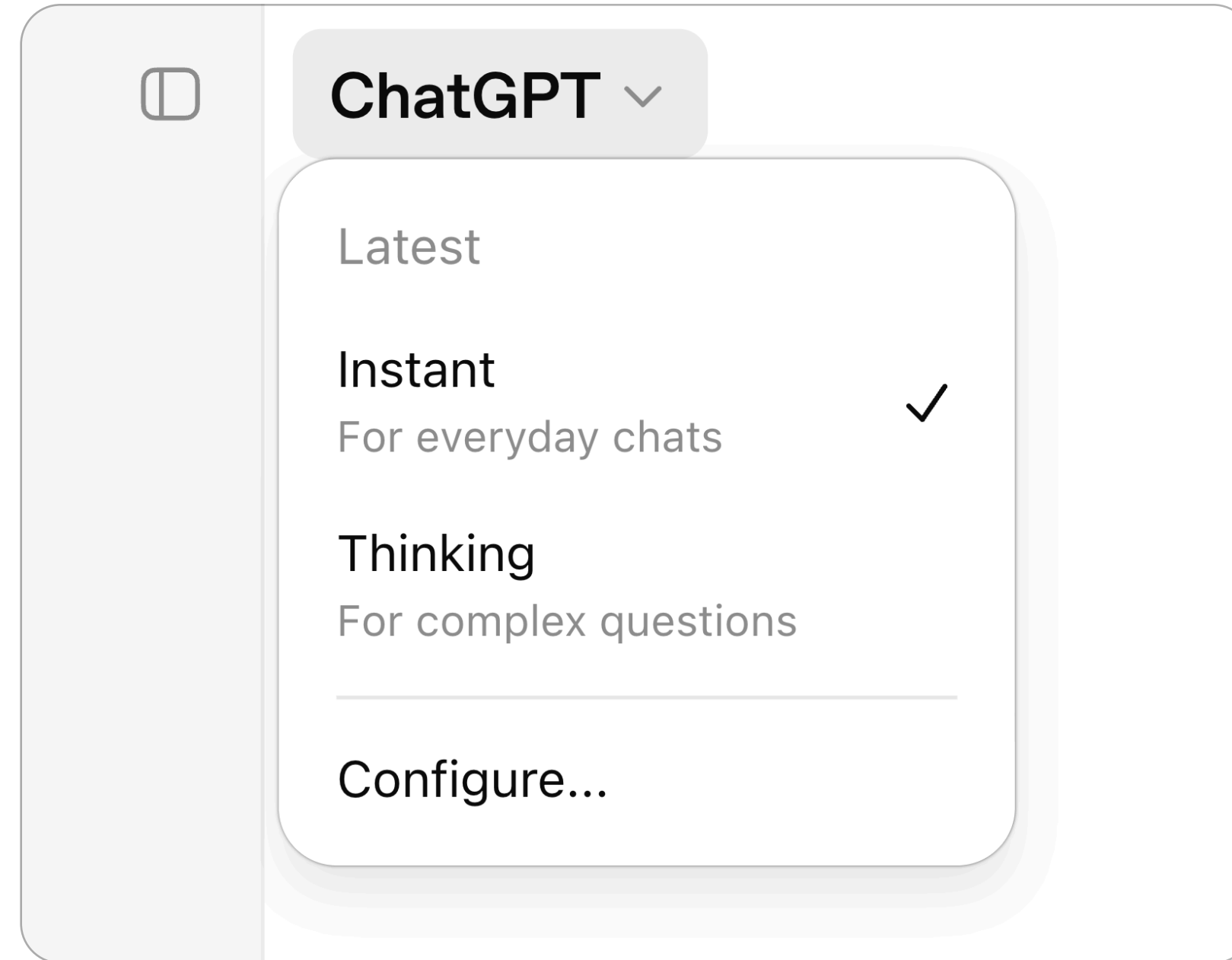
BUILDING ISN'T THE HARD PART

THE REAL PROBLEM: CLIENTS

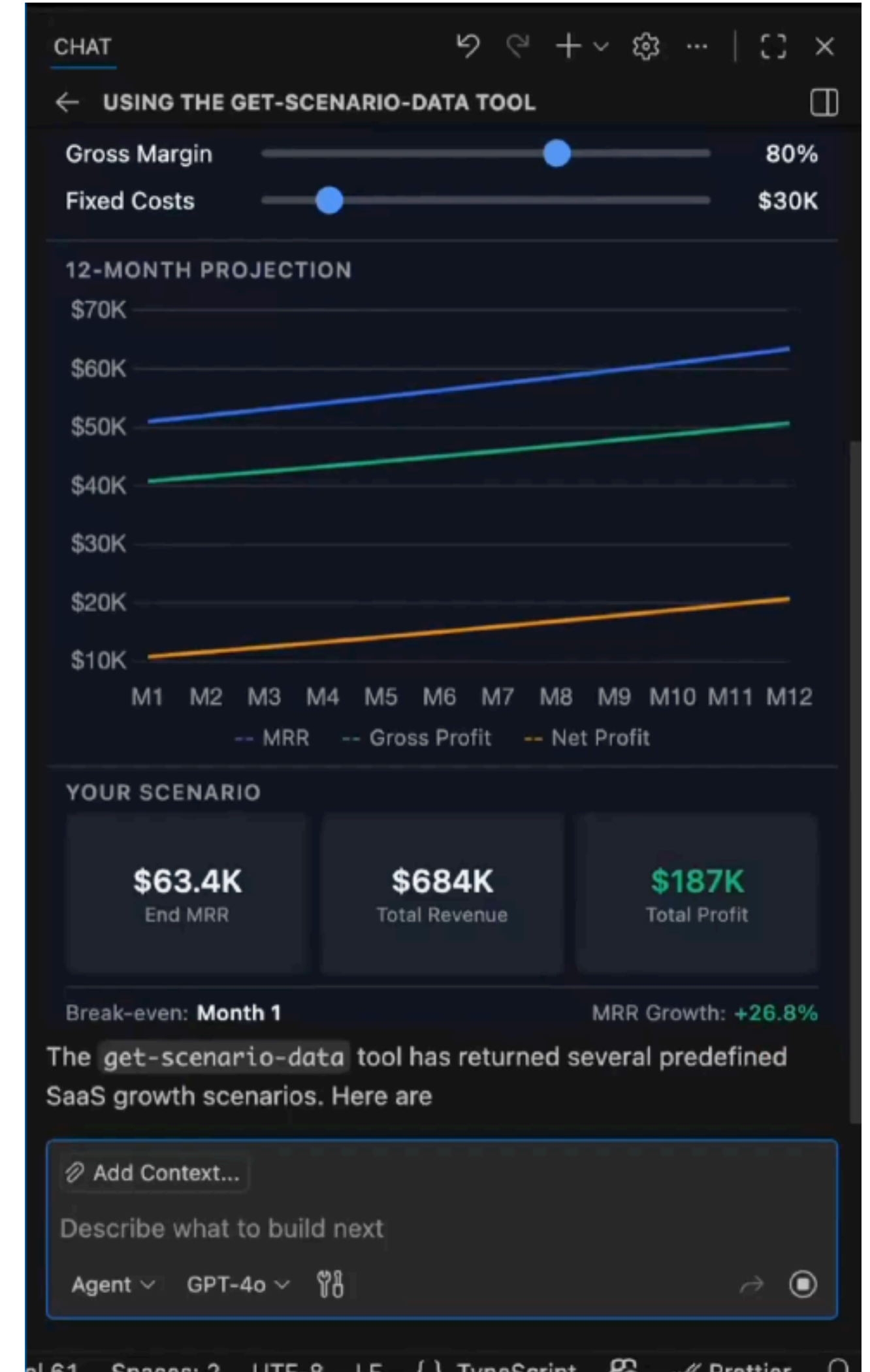
A CLIENT BUGS



B MODEL SMARTNESS



C LAYOUT



let's generate a slide deck based on what we said earlier



I'm turning the talk into a sharper conference deck now, using the angle we landed on: not just a flashy demo, but a practical live development loop with safety and production relevance.

I'm turning the talk we shaped into a conference deck outline and generating it in Figma Slides now.

Thought for 23s

{=} Generated Figma Slides deck

The first generation call timed out, so I'm retrying with a tighter prompt and the same substance.

{=} Generated Figma Slides deck

I tried generating it in Figma Slides twice, and the Figma deck tool timed out both times.

DEVELOPER MODE

Ask anything



Thinking



Figma



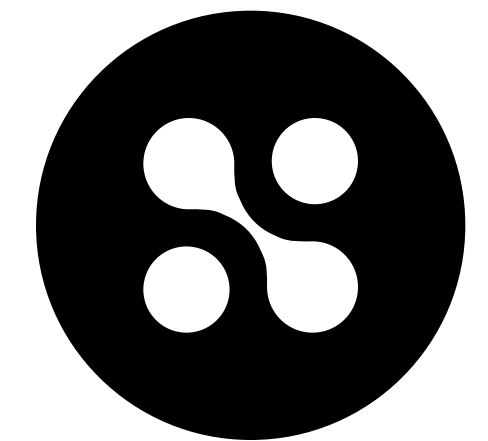
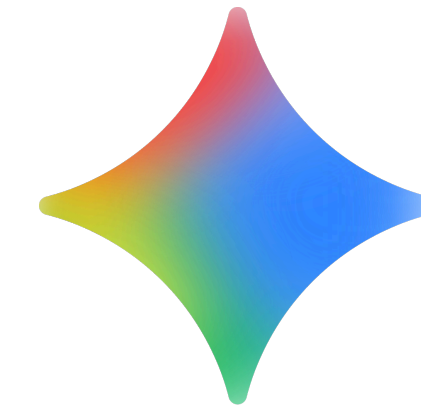
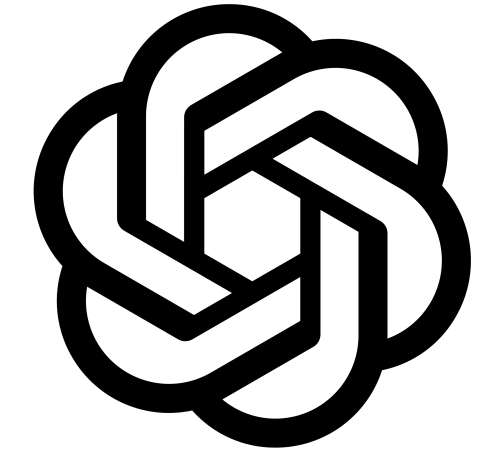
WHAT IF YOUR SERVER COULD TEST ITSELF?

WHY



2008

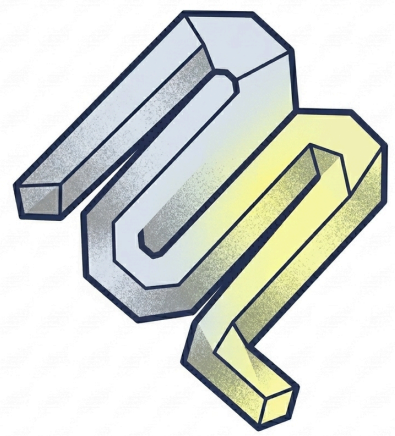
DEVELOPER TESTS AGAINST IE,
FIREFOX, SAFARI, CHROME



2026

~~AGENT~~
~~DEVELOPER~~ TESTS AGAINST CLAUDE, CHATGPT,
CURSOR, GOOSE. DIFFERENT MODELS,

WHAT



PROTOCOL

Tool returns wrong data, missing fields, type errors



BEHAVIOURAL

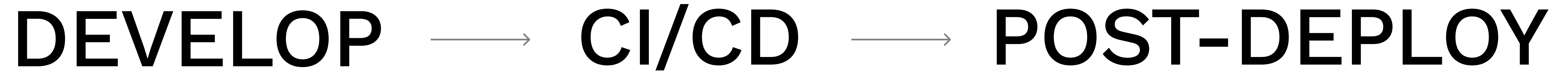
Model doesn't call the tool, calls it with wrong params, misinterprets the response



VISUAL

Widget doesn't render, layout broken, runtime JS errors

WHEN



HOW

/INSPECTOR

Let me create the tool

Edit server.ts

Now let me test it

showCart()

Let's test it in the chat

sendMessage()

getWidget()

EDIT THE SERVER

DYNAMICALLY CONNECT TO THE MCP SERVER ITSELF

INTERACT WITH EXTERNAL CHAT

CHATGPT

search-tools(1 args)

MCP-Apps Template

EN ❤️ 1

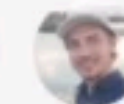
Lovely Little Fruit Shop

Tap a fruit to see details

Three fruit cards: an orange, a pineapple, and two cherries.

LIVE UPDATES THROUGH HMR AT THE PROTOCOL AND APP LEVEL

BRIDGE EVENTS RUNTIME JS ERRORS SERVER LOGS



Vibe your Claude Connectors

I want to build a cloud conn



Haiku 4.5



Projects 50

Templates

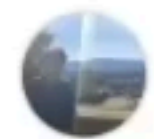
chore(deps): remove unused deps #282

Code

Open tonxxd wants to merge 4 commits into main from chore/deps

Conversation 3 Commits 4 Checks 33 Files changed 30

+5783 -1521



tonxxd commented 4 days ago

feat: add order tracking widget and refund estimation tool

Adds two new tools and a widget to the shop MCP server:

- `track-order` — returns real-time order status with carrier tracking info
- `estimate-refund` — calculates refund amount based on return reason and order age
- Order tracking widget with shipment timeline visualization

Also updates the `search-products` tool description to improve tool selection accuracy across models.

Testing

- Manufact cross-client evaluation passes on Claude Sonnet 4.5, GPT-4o, GPT-4.1
- Widget renders correctly in Claude Desktop and ChatGPT



feat: add order tracking widget and refund estimation tool

f61c1ae



Manufact bot commented 4 days ago · edited

Reviewers

Suggestions



Pederzh

Request

At least 1 approving review is required to merge this pull request.

Still in progress? [Convert to draft](#)

Assignees

No one—[assign yourself](#)

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

Successfully merging this pull request may close

- Back to Org
- Overview**
- Chat
- Primitives
- Tools 6
- Prompts 0
- Resources 2
- Deployments
- Deployments
- Logs
- (x) Environment Variables
- Analytics
- Analytics
- Observability
- Testing
- Testing Beta
- Settings
- Settings
- Domains

shop-mcp

Deployed from GitHub repository tonxxd/shop-mcp

Tools 6

Prompts 0

Resources 2

Observability
Events over last 7 days



Server Information

Title	shop-mcp	Name	shop-mcp	Icons	
Version	1.0.0	Website URL	-		

Repository

[View](#)

Active Deployment

[a3d538e](#) Initial commit from temp...

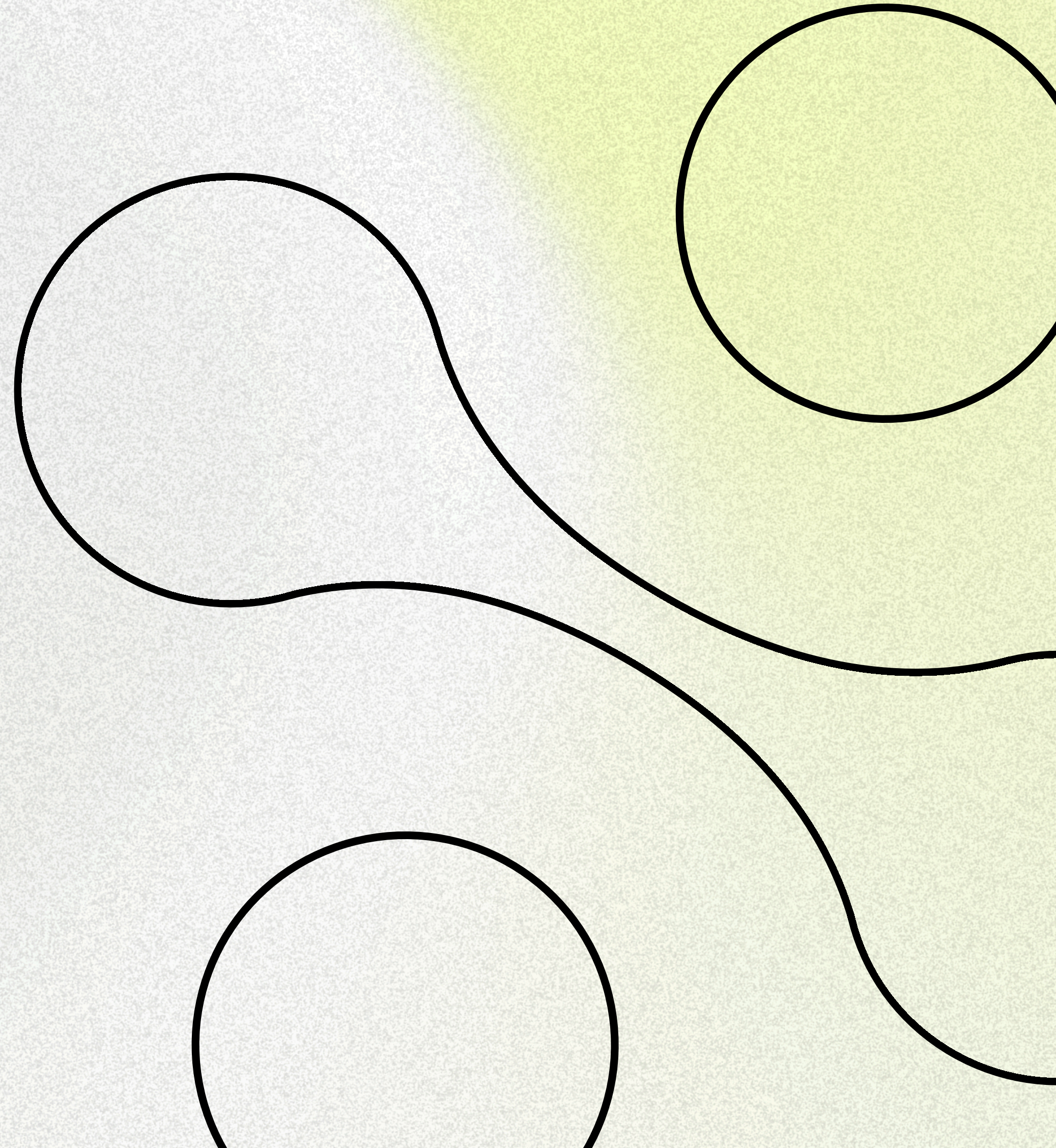
Connection Status

● Connected

Tags

+ Add tag

WHO WE ARE



MCP-USE FRAMEWORK

The fullstack MCP framework to build MCP Apps for ChatGPT/Claude and MCP Servers for AI Agents.



★ 9,600 GitHub stars

↓ 5,000,000 downloads

QUICKSTART

- ***npx create-mcp-use-app***
- Scaffold a complete an MCP server project with all necessary configuration for MCP Apps
- Template to select: **MCP Apps**

```
# 1. Create your project
npx create-mcp-use-app@latest

# 2. Enter the directory
cd my-server

# 3. Start development (opens inspector automatically)
npm run dev


# 4. Open the inspector
# → http://localhost:3000/inspector
```



ARCHITECTURE

- **Server:** defines tools + resources
- **Widget:** React component in `resources/` folder, renders in the client
- **Protocol:** MCP handles the communication automatically

```
Project structure:
my-server/
├── public/
├── resources/
│   └── product-search-result/
│       └── widget.tsx
├── index.ts (server entry point)
├── package.json
├── tsconfig.json
└── README.mdr
```

Made with  by mcp-use

INSTALL THE SKILL

- Adds the **mcp-apps-builder** skill to your Agent
- Gives you instant scaffolding for tools, widgets, and resources
- Knows all the mcp-use SDK patterns

skills / mcp-use / mcp-use / mcp-apps-builder

mcp-apps-builder

```
$ npx skills add https://github.com/mcp-use/mcp-use --skill mcp-apps-builder
```

SKILL.md

IMPORTANT: How to Use This Skill

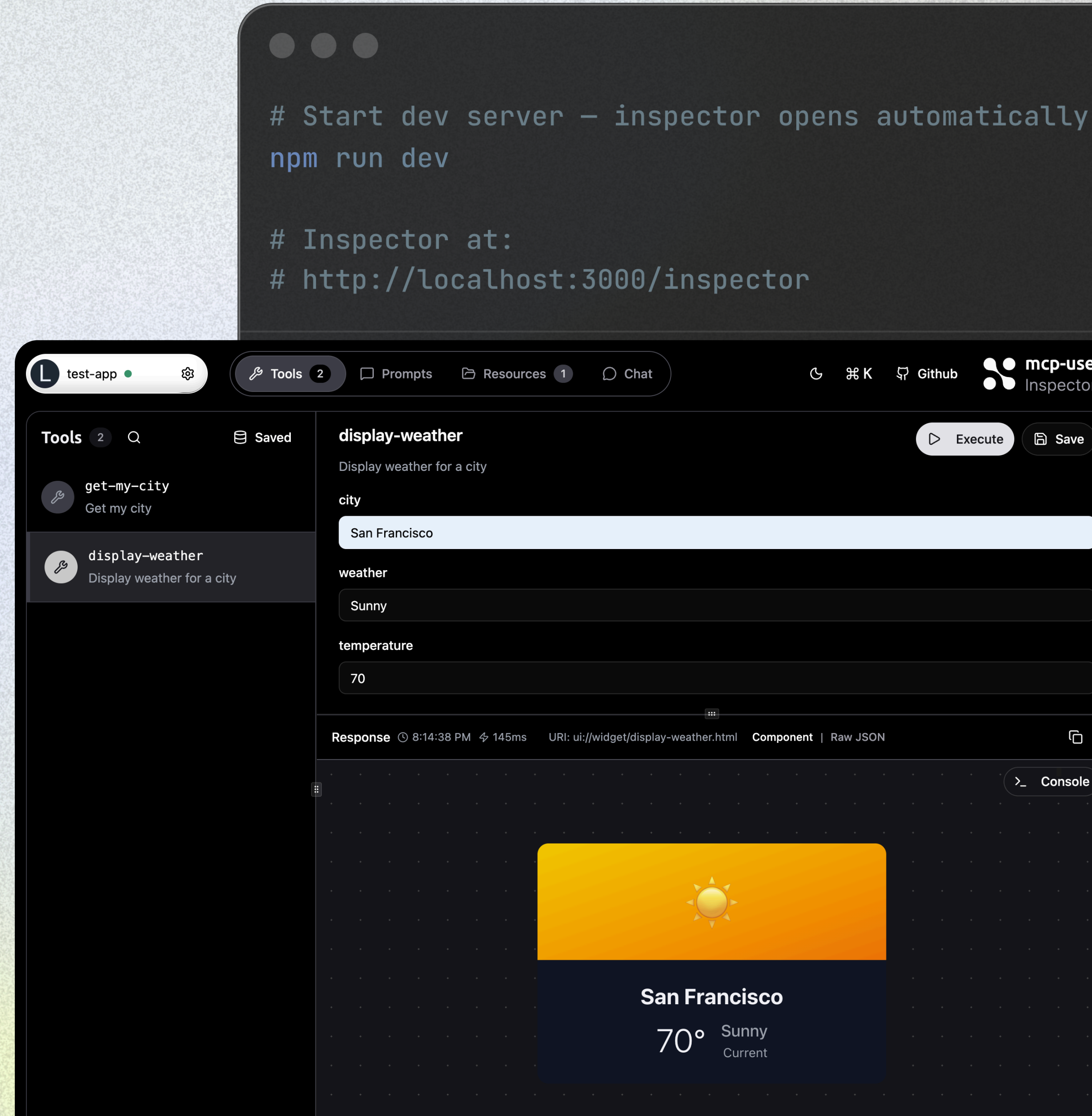
This file provides a NAVIGATION GUIDE ONLY. Before implementing any MCP server features, you MUST:

1. Read this overview to understand which reference files are relevant
2. **ALWAYS** read the specific reference file(s) for the features you're implementing

<https://skills.sh/mcp-use/mcp-use/mcp-apps-builder>

PREVIEW WITH INSPECTOR

- List all MCP tools and resources
- Call any tool with test input
- See widget previews rendered live
- Hot reload: change code, see results instantly
- Also available online at inspector.mcp-use.com



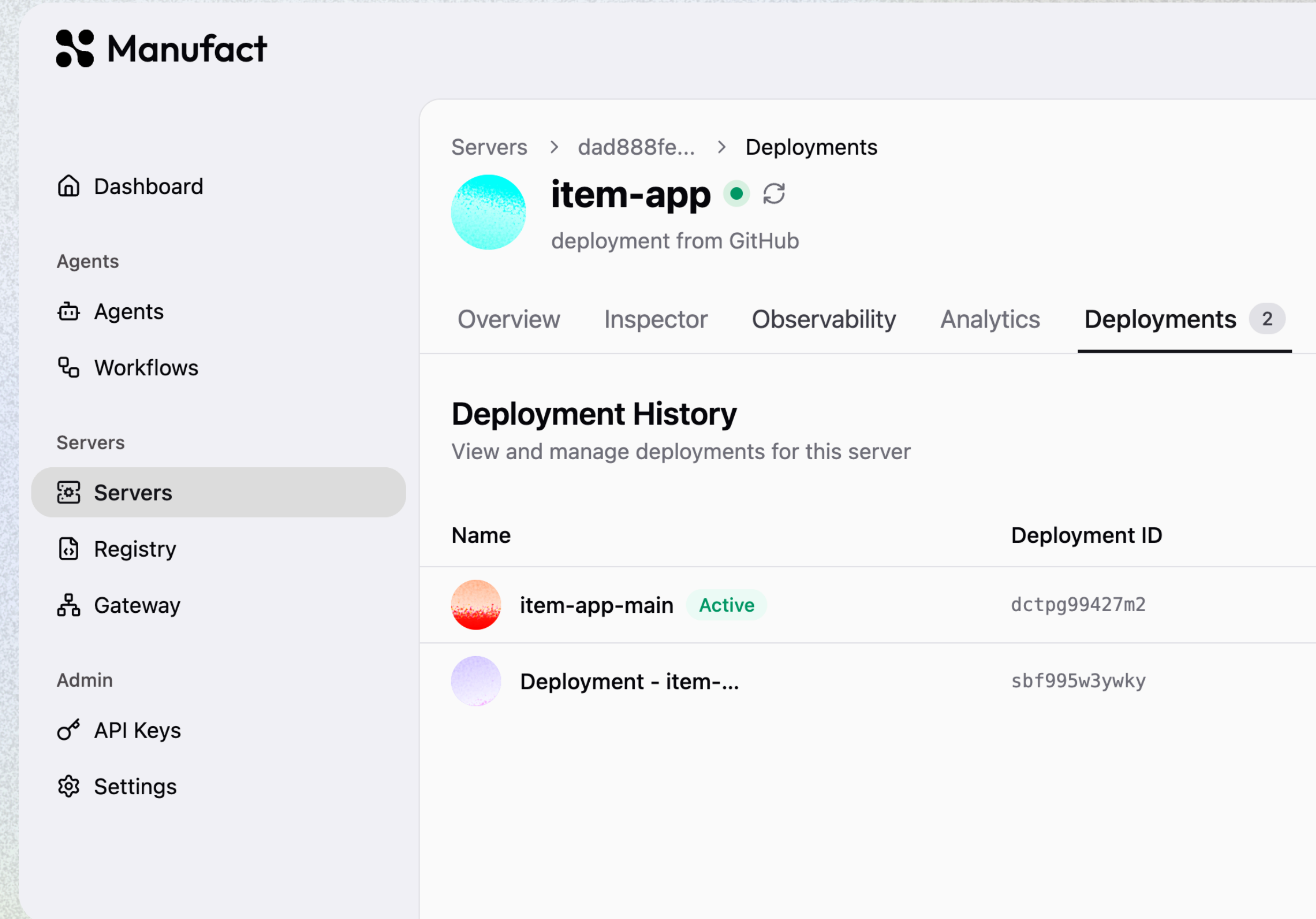
DEPLOY TO MANUFACT CLOUD

1. Go to manufact.com

2. Connect your GitHub repo

3. Push to main → auto-deploys

OR: `npx @mcp-use/cli deploy`





EMAILS

founders@mcp-use.com

pietro@mcp-use.com

luigi@mcp-use.com

enrico@mcp-use.com



github.com/mcp-use/mcp-use