Google Developer ... · Follow publication

Google Antigravity: How to add custom MCP server to improve Vibe Coding

5 min read · Nov 20, 2025

Tarun Jain Follow

Listen Share More

Google just announced Gemini 3 Pro and the Antigravity release. I am actually more excited about Antigravity because of **Varun Mohan**. I have been a big fan of Windsurf, and I had a feeling from the start that once Varun joined the team, something amazing would arrive.

> *And here we are with Antigravity.*

> I have been experimenting with it for the past few hours and here are my thoughts:

What is Antigravity?

Google Antigravity is a new Agentic development platform from Google that changes how software is built by letting developers work at a higher task-focused level. With the reasoning and Agentic coding abilities of **Gemini 3**, Antigravity allows AI Agents to plan, code, and validate complete software tasks on their own.

It includes an Agent manager dashboard, a VS Code-style editor, and deep browser connectivity, letting Agents interact with web apps directly for testing and validation. Instead of being just another coding tool, it becomes a true development partner that improves productivity by managing multiple projects and tasks in parallel with autonomous workflows.

> Download now: https://antigravity.google/download

Important notes to improve your Vibe Coding with Antigravity

If you want to start with Antigravity, first install it and pick your preferred IDE layout. After setup, move to planning before writing any code.

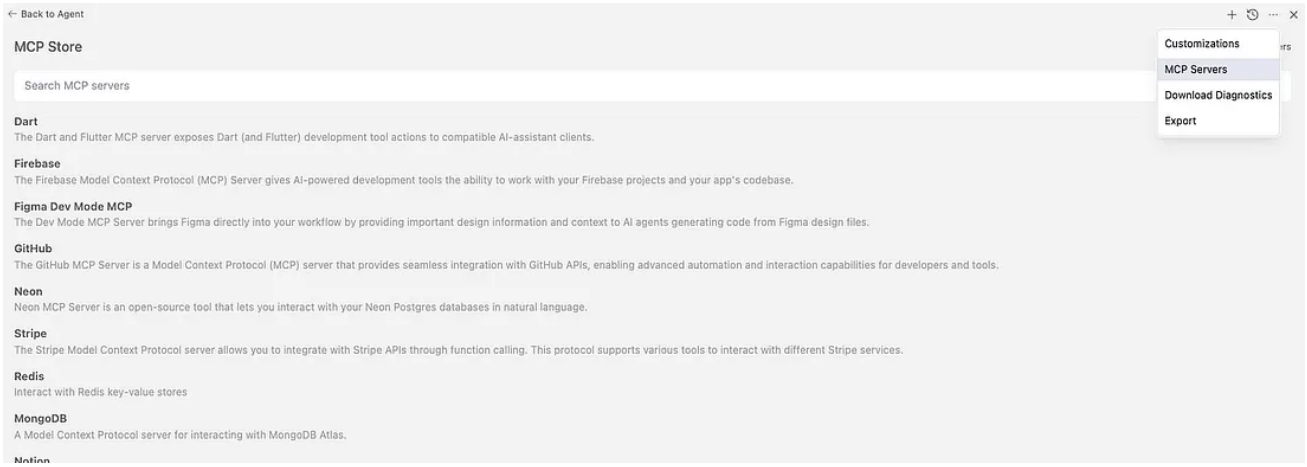
1. Begin by defining your task and the features you need. Then prepare an implementation plan and workflow. If you already have clarity, write it yourself. If not, simply describe the goal and let Gemini 3 generate the **Implementation Plan**, **Task**, and **Walkthrough** files.
2. **IMPORTANT**: DO review the **Implementation plan**, **Task list**, and **Walkthrough**. If there is any task related to testing, remove that. Do tell the Agent not to run tests automatically.

3. Do the testing manually because **automated testing from the Agent can consume more tokens than required**.
4. When Vibe coding, the key stage is implementing features. Avoid asking the Agent to build all features in a single run. If you have more than 5 features, begin with the first one and then move step by step.
5. After completing the first session, ask Gemini to generate a **SUMMARY.md** of the conversation. Save this summary since you can reuse it for future sessions without repeating the entire process.
6. Once you have two features ready and you have already reviewed the **Implementation plan** and workflow, start a new task using the saved SUMMARY as context and follow the same process.
7. Now for further improvements, add **Custom MCP servers** that keep you updated with documentation and SDK references for your tech stack. MCP Servers can help fetch context for better code generation and the latest.

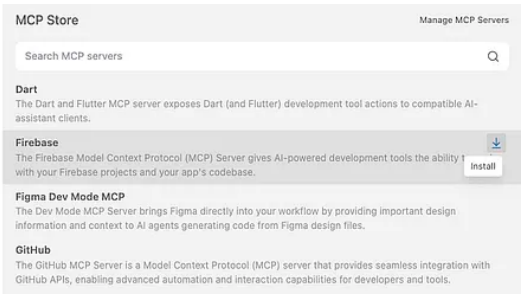
> Lets see how to add your own custom MCP Server in Google Antigravity.

How to add a Custom MCP server in Google Antigravity

1. Click on Agent session and select the "..." dropdown at the top of the editor's side panel. And select MCP Servers, you will find the MCP Store.



2. To add a custom MCP server, select **Manage MCP Servers** at the top of the MCP store and then click on View raw config in the main tab.



3. Modify the mcp_config.json with your custom MCP server configuration. Note for Vibe coding, having these 3 MCP servers can be handy:

- Context-7: Up-to-date Code Docs for the tech stack your app needs.
- Sequential Thinking: For dynamic and reflective problem-solving through a structured thinking process.
- Retriever-based MCP: Save and retrieve the code that works for future purposes. for example: **Qdrant MCP Server**.

> NOTE: **Sequential Thinking is already in MCP Store, just click on Install.**

mcp_config.json:

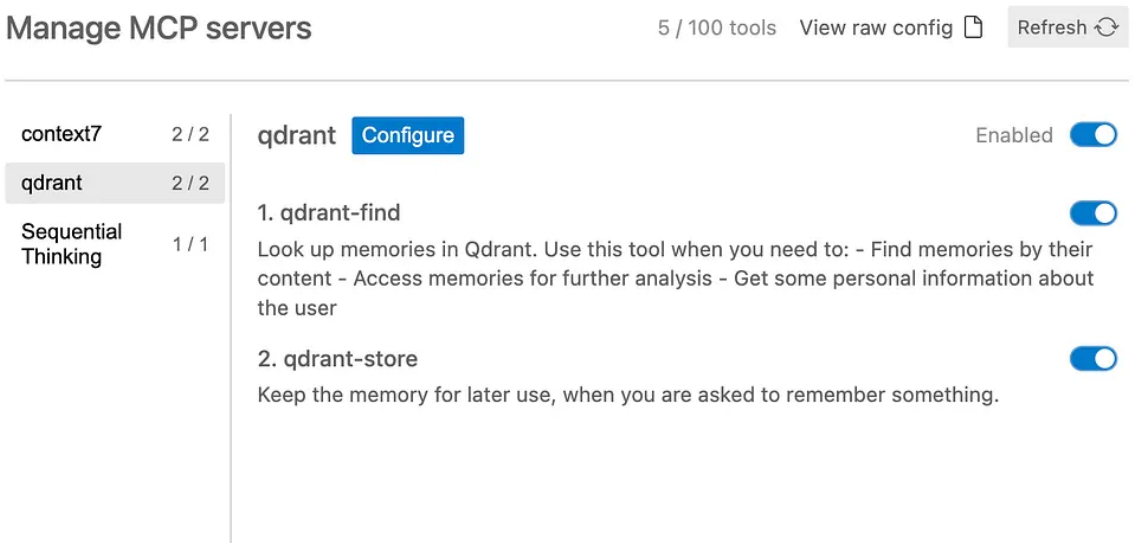
> *Syntax for npx and uvx based MCP*

```
{
    "mcpServers": {
        "sequential-thinking": {
            "command": "npx",
            "args": [
                "-y",
                "@modelcontextprotocol/server-sequential-thinking"
            ]
        },
        "context7": {
            "command": "npx",
            "args": [
                "-y",
                "@upstash/context7-mcp",
                "--api-key",
                "<replace-with-your-Context7-API-Key>"
            ]
        },
```

```
        "qdrant": {
            "command": "/Users/tarunjain/.local/bin/uvx",
            "args": [
                "/Users/tarunjain/Desktop/mcp-server-qdrant"
            ],
            "env": {
                "QDRANT_URL": "<replace-with-your-Qdrant-Endpoint>",
                "QDRANT_API_KEY": "<replace-with-your-Qdrant-API-Key>",
                "COLLECTION_NAME": "vibe-coding",
                "EMBEDDING_MODEL": "sentence-transformers/all-MiniLM-L6-v2"
            }
        }
    }
}
```

> Get your Qdrant URL and API key after creating a free cluster on: **Qdrant Cloud**
>
> Get your Context-7 API key from here: **Context 7 Dashboard**

4. Now, come back to Managed MCP servers and click Refresh.



> We have 5**tools across 3 MCP Servers.** Context-7 have 2 tools, Qdrant have 2 tools and Sequential Thinking have one tool.

If you are new to MCP, watch my video where I use the Qdrant MCP server with Claude Desktop and explain what MCP is, why you need it, and how it works.

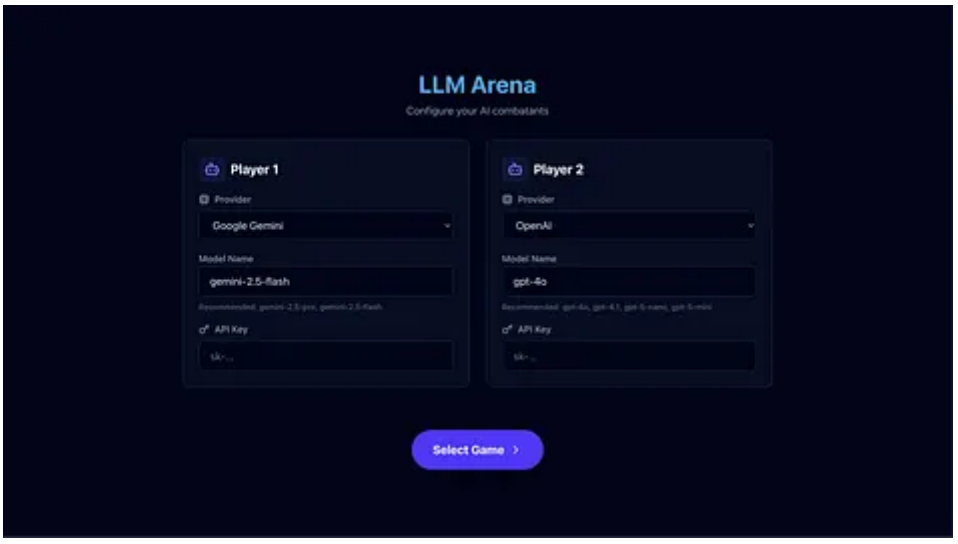
Demo

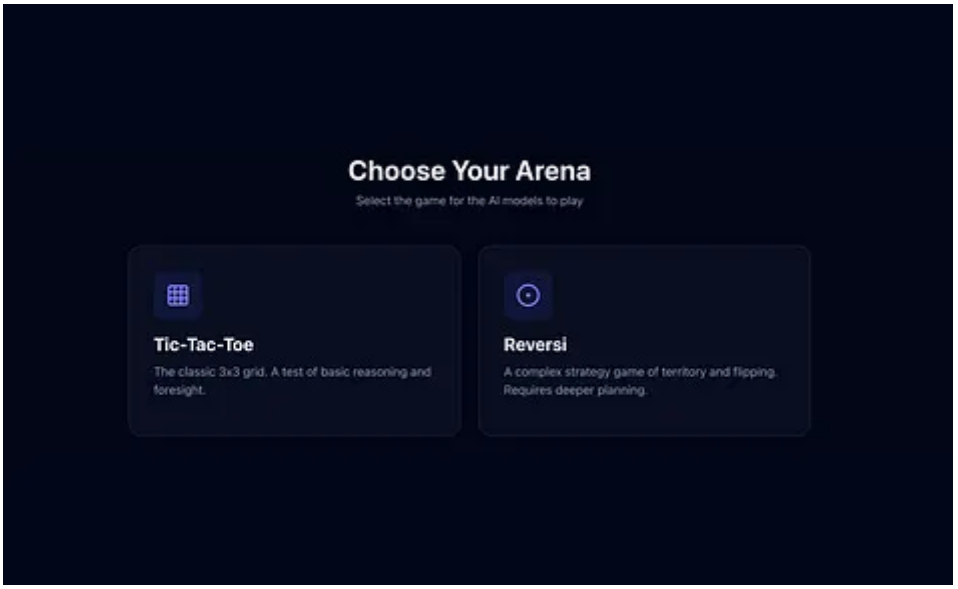
> Prompt

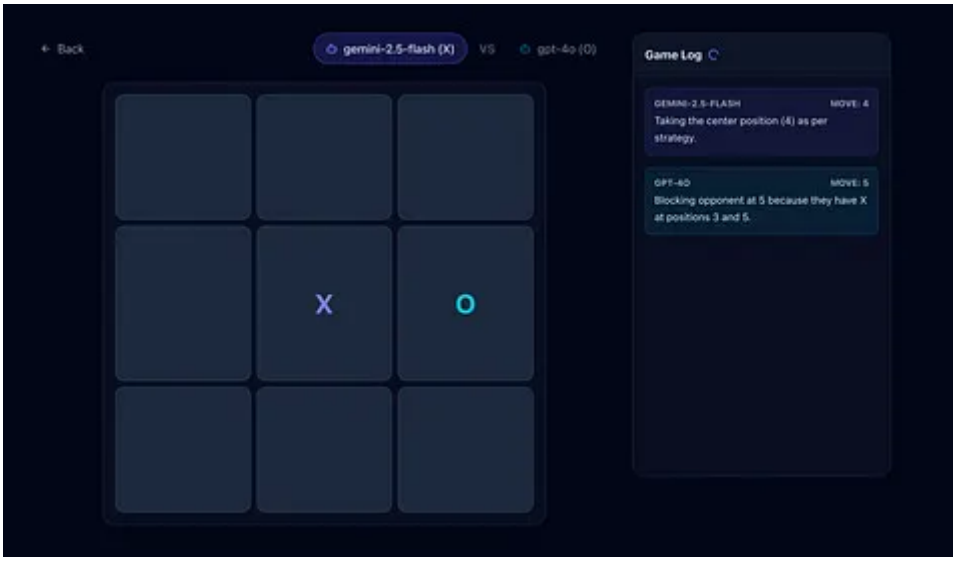
```
Plan: To build a 2 player game, but the catch is, the 2 players here are the

Scene-1: Chose 2 players. we need to have options selectbox to pick the LLM
Each LLM choice needs to have to options: Enter API key and Choose the model

Scene-2: Now I need to have 2 games: Tic-Tac-Toe and Reversi: Both 2 player

Scene-3: Once the game is selected, in scene 3 I need the visuals, to check

For LLM inference use LiteLLM:

import os
from litellm import completion

os.environ["OPENAI_API_KEY"] = "your-api-key"
response = completion(
    model = "gpt-4o",
    messages=[{ "content": "Hello, how are you?","role": "user"}]
)
os.environ['GEMINI_API_KEY'] = ""
response = completion(
    model="gemini/gemini-pro",
    messages=[{"role": "user", "content": "write code for saying hi from Lit
```

```
)

os.environ["ANTHROPIC_API_KEY"] = "your-api-key"

messages = [{"role": "user", "content": "Hey! how's it going?"}]
response = completion(model="claude-opus-4-20250514", messages=messages)
print(response)
```

> First start with Tic-Tac-Toe and then Reversi. One game at a time.
>
> App Screenshot







🔗 **GitHub Repo** — LLM Game Vibe Coded: https://github.com/lucifertrj/llm-game-battle

VIDEO TUTORIAL COMING SOON. Subscribe to YouTube

https://youtube.com/@aiwithtarun

Mcp Server | Antigravity | Google | Agents | AI Agent

Follow

Published in Google Developer Experts

35K followers · Last published 3 days ago

Experts on various Google products talking tech.

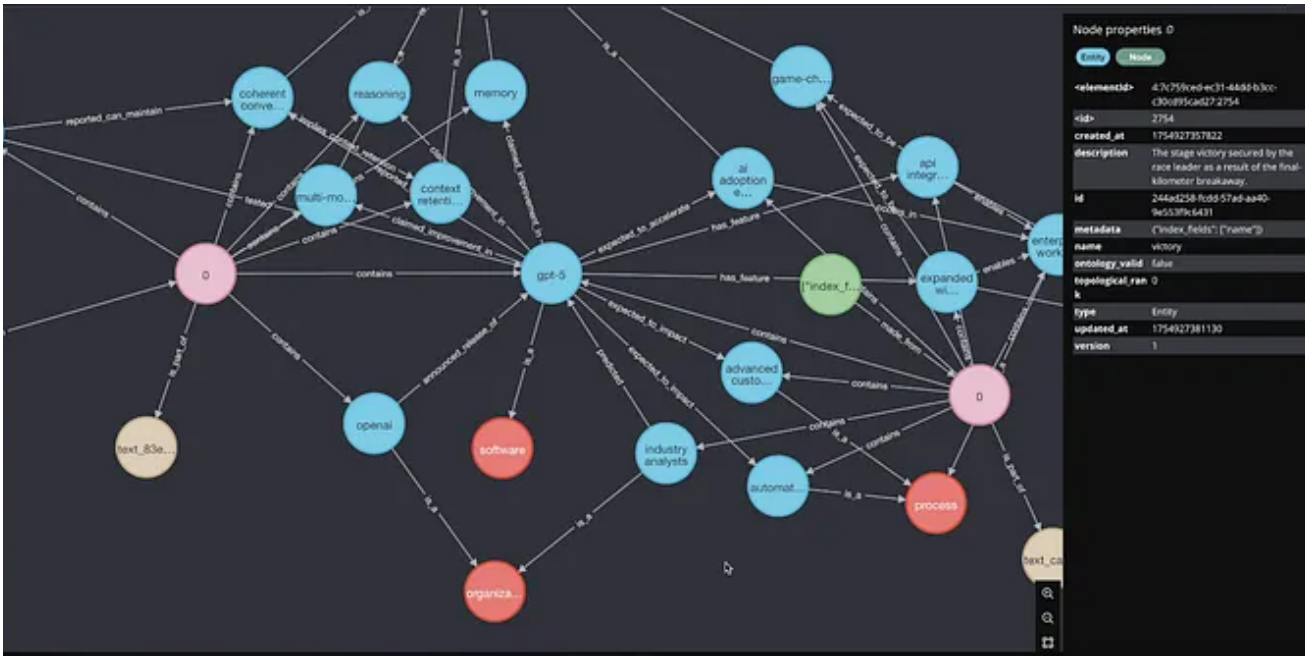
Follow

Written by Tarun Jain

238 followers · 3 following

Youtube: AIWithTarun || ML @AIPlanet || GSoC'24 RedHen Lab ||GSoC'23 @caMicroscope || GDE in ML

More from Tarun Jain and Google Developer Experts



In Google Developer Experts by Tarun Jain

Implementing Long Term Memory for Google ADK using Cognee

While Large Language Models have demonstrated impressive capabilities in generating text and solving reasoning problems, their inability...

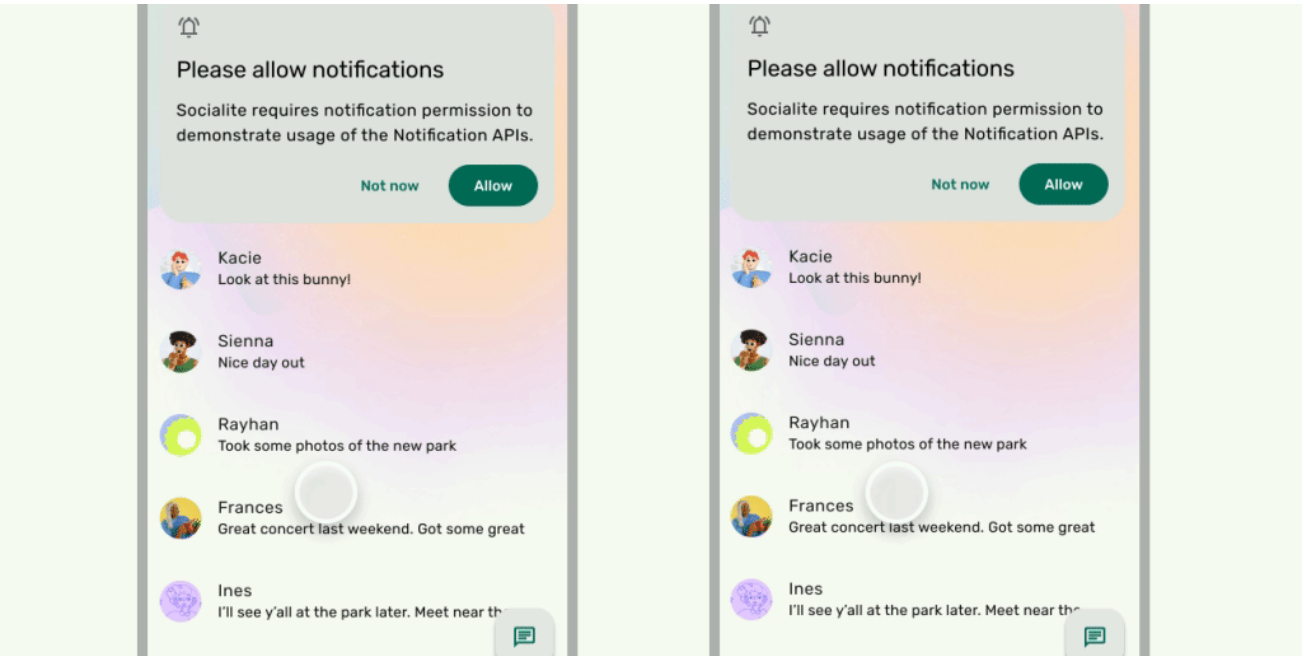
Dec 31, 2025 73 2

In Google Developer Experts by Sasha Denisov

On-Device Function Calling with FunctionGemma

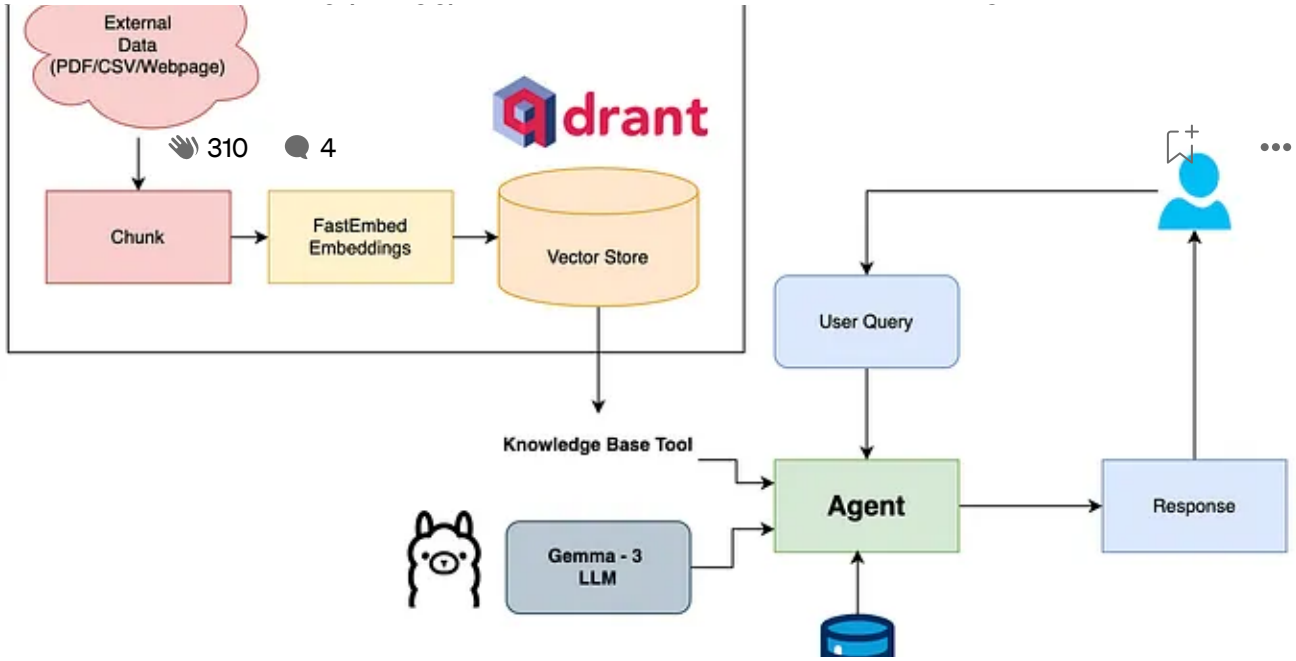
A Practical Guide to preparing and using Google's smallest function calling model for Mobile and Web Developers

Dec 29, 2025 102 1



In Google Developer Experts by Qamar A. Safadi

Edge-to-Edge Is No Longer Optional — Android 16 Migration Guide.



In AI Planet by Tarun Jain

Building a Local Agentic RAG System using Gemma-3, FastEmbed and Qdrant Vector database

In this article, we will explore how to build a 100% local agentic RAG system using open-source libraries. This system allows you to create...

Mar 13, 2025 91 3

See all from Tarun Jain

See all from Google Developer Experts

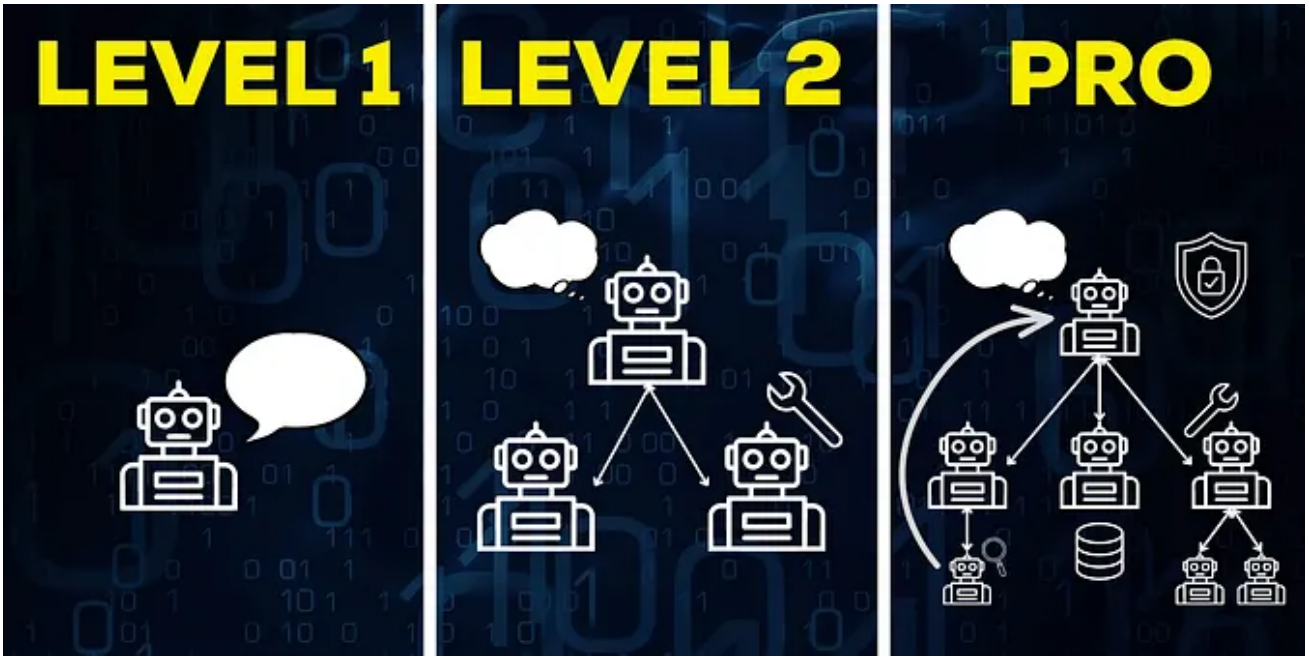
Recommended from Medium

In Artificial Corner by The PyCoach

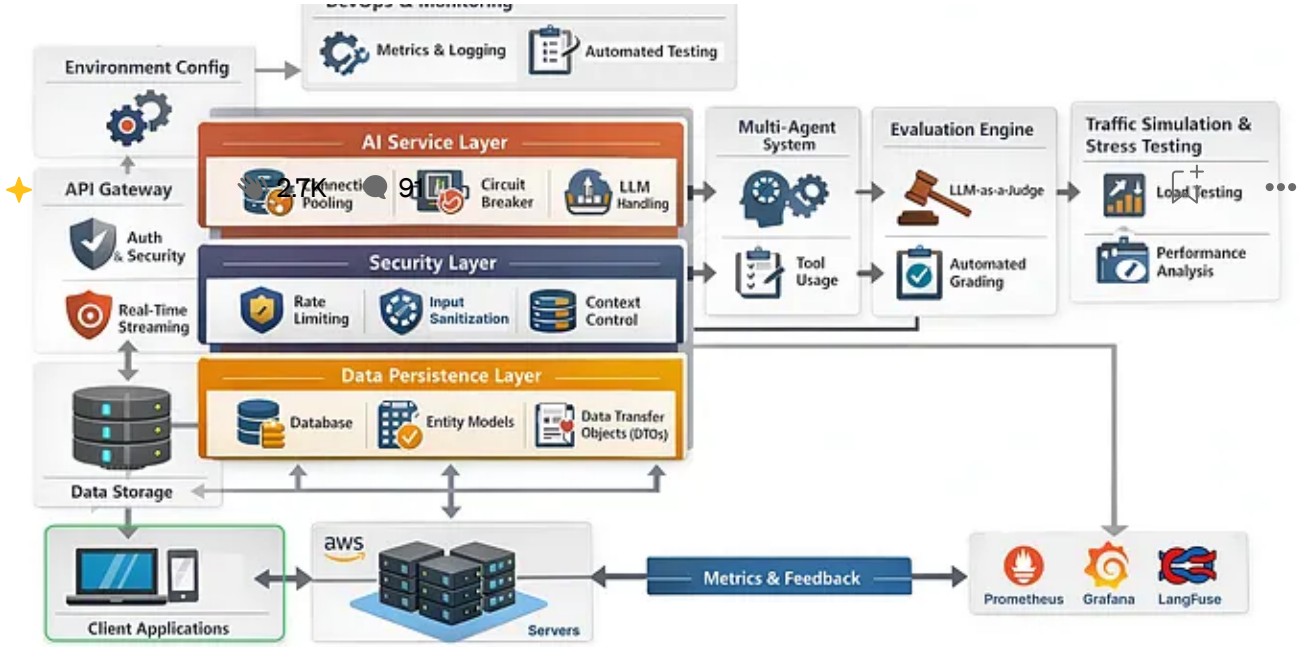
The Best AI Tools for 2026

If you're going to learn a new AI tool, make sure it's one of these

Dec 1, 2025 4K 210



In Data Science Collective by Marina Wyss - Gratitude Driven

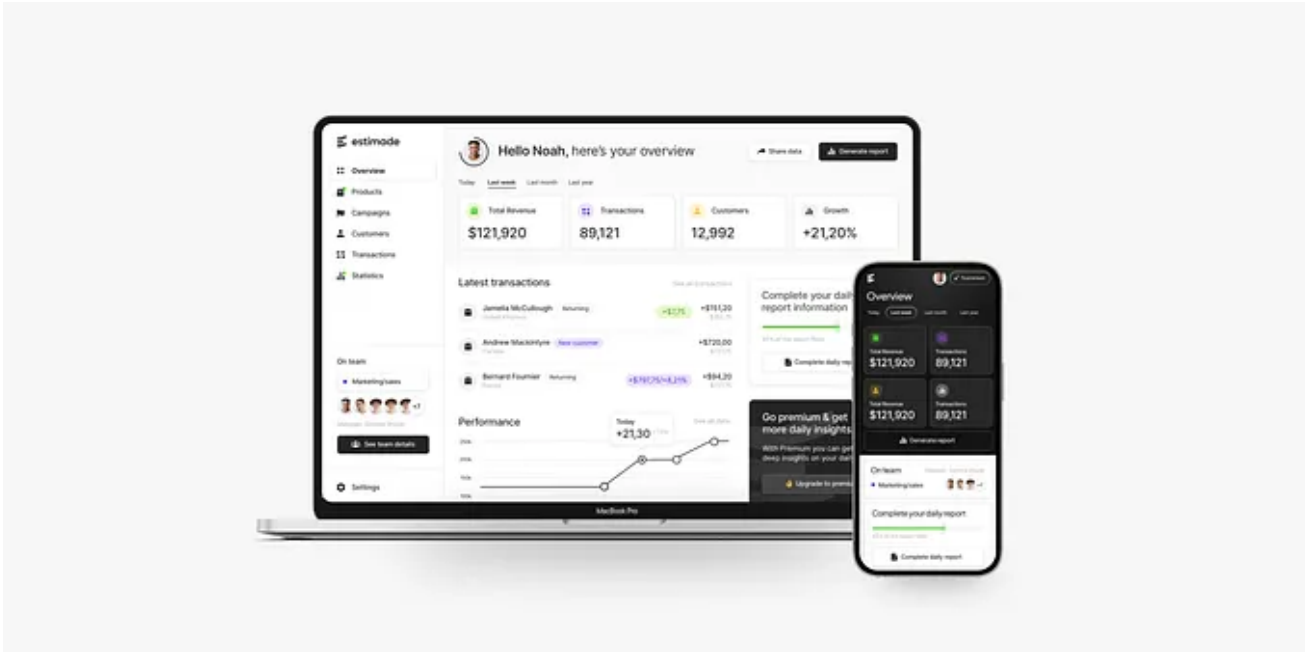


In Level Up Coding by Fareed Khan

Building the 7 Layers of a Production-Grade Agentic AI System

Service Layer, Middleware, Context Management and more

Dec 19, 2025 1.5K 21



Michal Malewicz

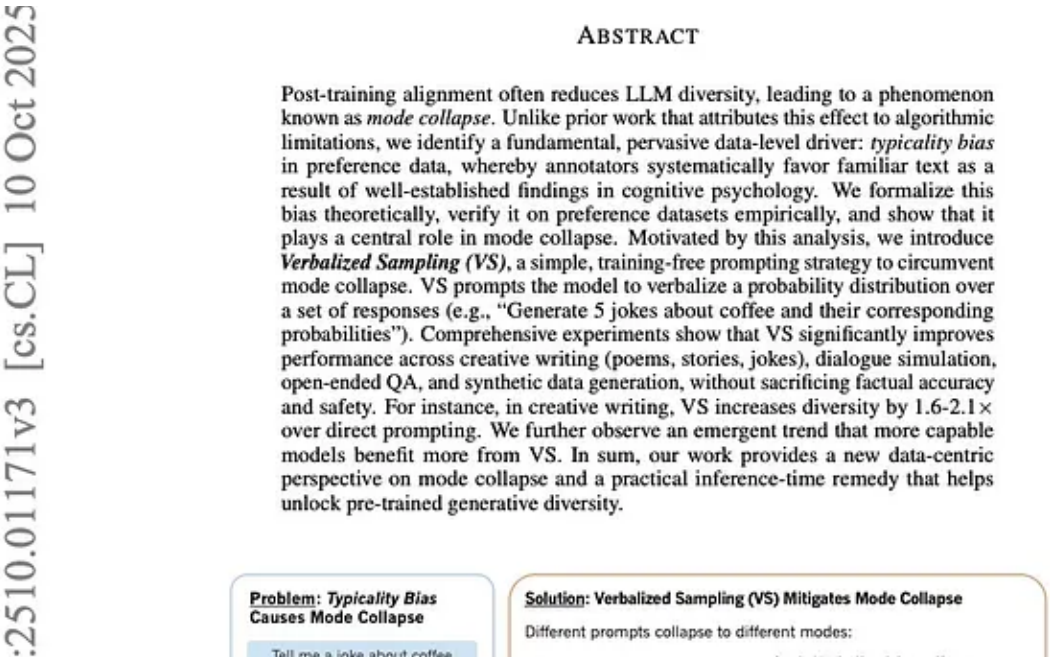
The End of Dashboards and Design Systems

Joe Njenga

I Tried New Claude Code Ollama Workflow (It's Wild & Free)

Claude Code now works with Ollama, which takes the game to the next level for developers who want to work locally or need flexible model...

4d ago 652 10



In Generative AI by Adham Khaled

Stanford Just Killed Prompt Engineering With 8 Words (And I Can't Believe It Worked)

ChatGPT keeps giving you the same boring response? This new technique unlocks 2× more creativity from ANY AI model — no training required...

Oct 20, 2025 23K 587

See more recommendations