Gemini

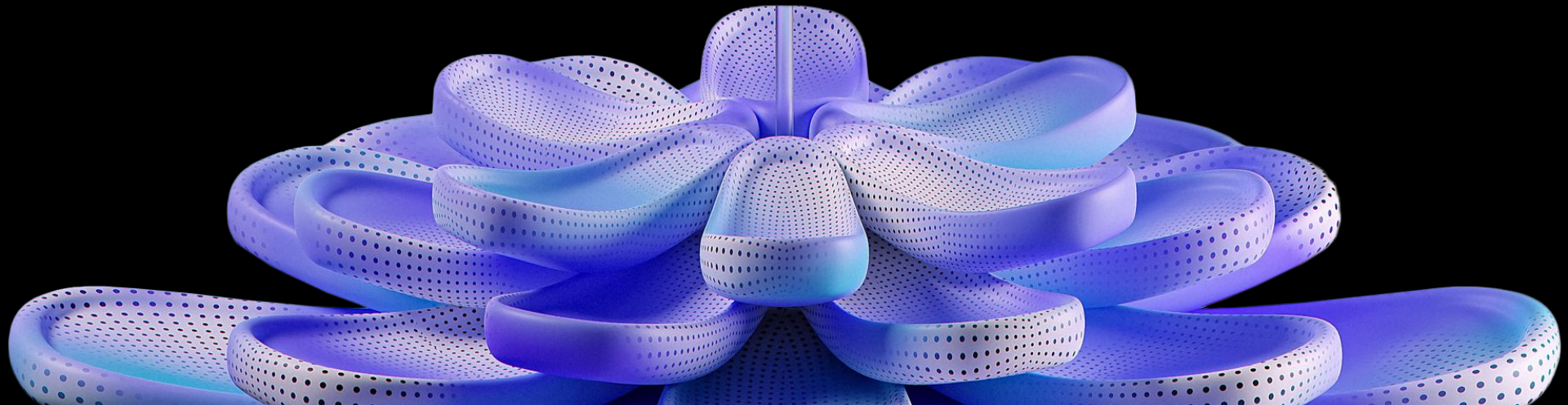
Vibe Coding 101

Mahin Sheth

Data and AI Engineer

Google

The opportunity with AI is as big as it gets.



Gemini 3
Bring any idea to life

In Today’s Session

1. What is Vibe Coding?
2. Popular Google Tools
3. Demo Time
4. From Idea to a Prototype

What is Vibe coding?

PRIVILEGED & CONFIDENTIAL

The Evolution of AI Development

Code Generation

The "Assistant" model. Tools like Copilot that autocomplete syntax and write small functions.

- **Goal:** Speed up typing.
- **Scope:** Single lines or blocks.
- **Vibe:** "I'm driving, AI is the GPS."

Vibe Coding

The "Creator" model. Creating entire apps via natural language prompts and iteration.

- **Goal:** Immediate results.
- **Scope:** Full files or prototypes.
- **Vibe:** "Just make it work, I don't care how."

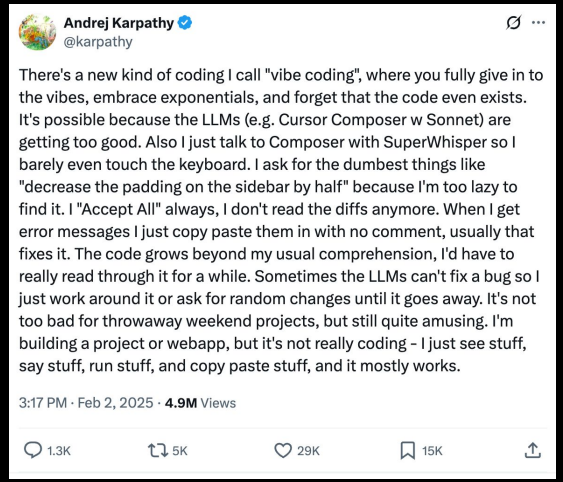
Have you ever had a great idea for an app but didn't know how to code?

What is Vibe Coding?

- AI-assisted development process.
- It's about describing your idea in plain English and letting AI be your coding partner.

Who coined this term?

Vibe coding is an artificial intelligence-assisted software development style popularized by Andrej Karpathy in early 2025



Myth Busting

Myth: You need to be a programmer to build an app.

Reality: With tools like Gemini and AI Studio, you can create a functional prototype without writing a single line of code.

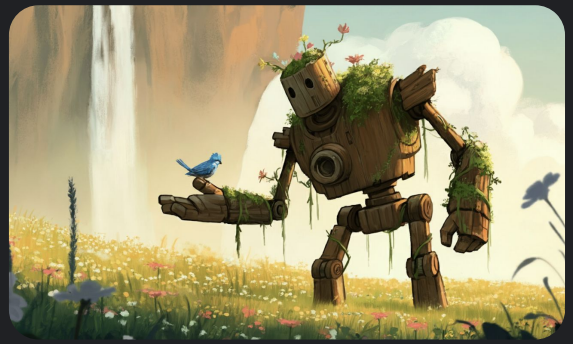
Myth: AI does everything, and you just watch.

Reality: Vibe coding is a partnership. Your role is to guide the AI, test the results, and provide feedback.

The case for Vibe Coding

Amazing way to get in tune with the pace of progress in AI

Every time I start a new project, I'm staggered by how good AI has become.



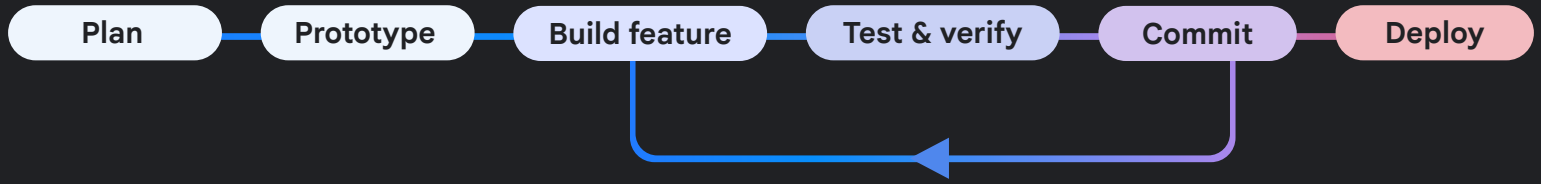
Fun, useful and addictive - the only limit is your patience

...and being able to finish one project before moving on to another.

Teaches you a new way to use your brain and see the world

This is also very useful in a work context.

How Vibe Coding Works



Plan out your idea

Check viability, plan features, understand end-to-end process

Choose your AI tool

Depends on required technical complexity and freedom, and on your skills and preferences

Build your features

By using natural language, testing features after each addition, and having lots of patience

Share your work

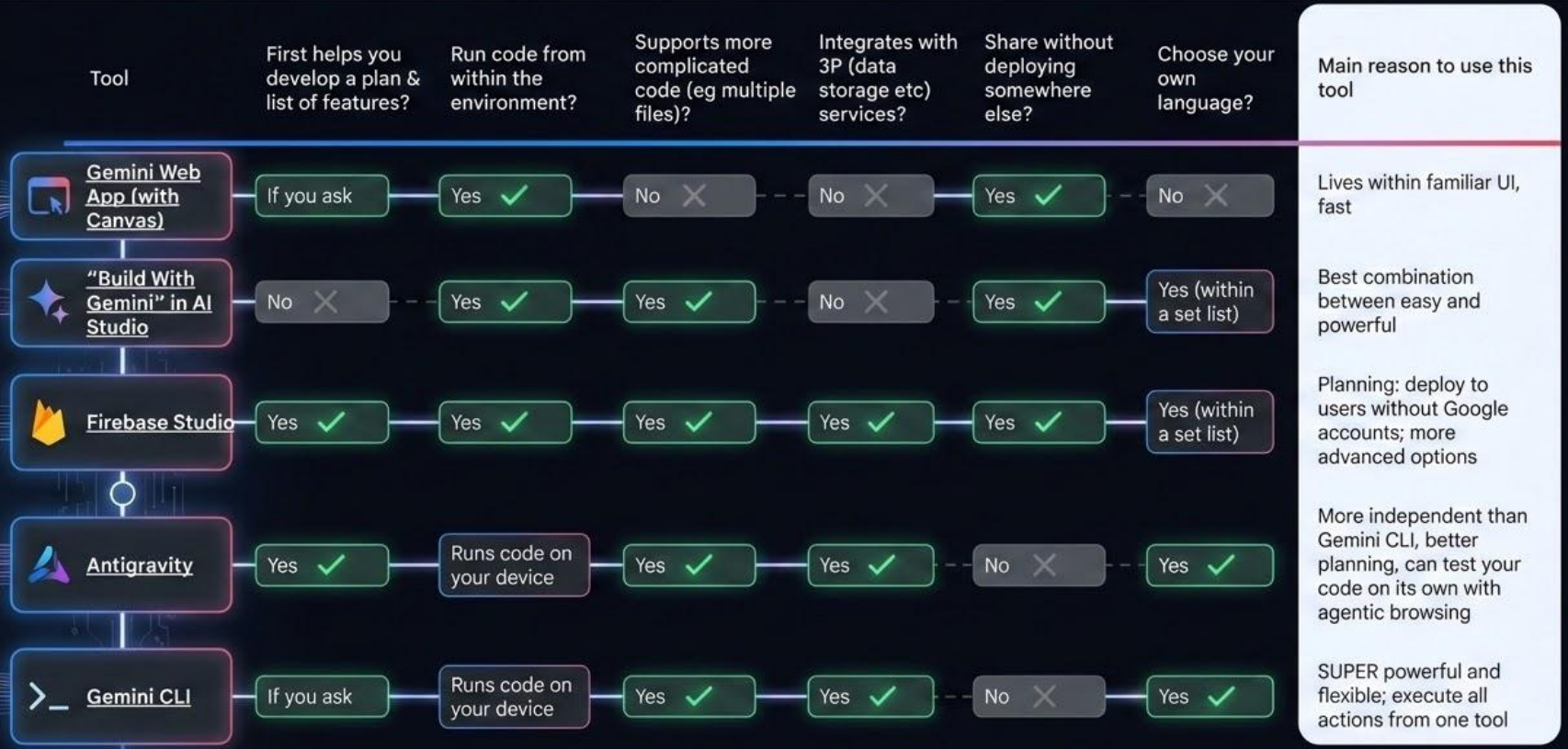
Options to export, share or deploy your work will vary by tool, as will the amount of work required

Some Popular Google Tools

PRIVILEGED & CONFIDENTIAL

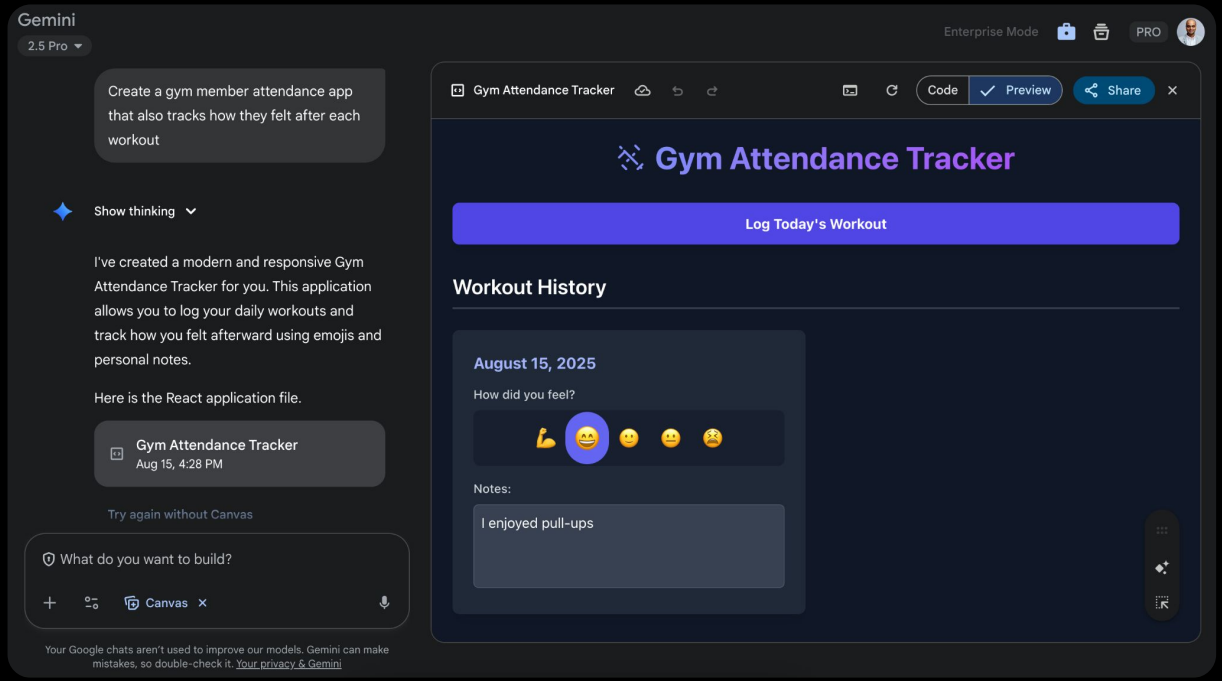
Google's suite of vibecoding tools

Shortlist, ranked from simple to more complex and powerful



Gemini Canvas

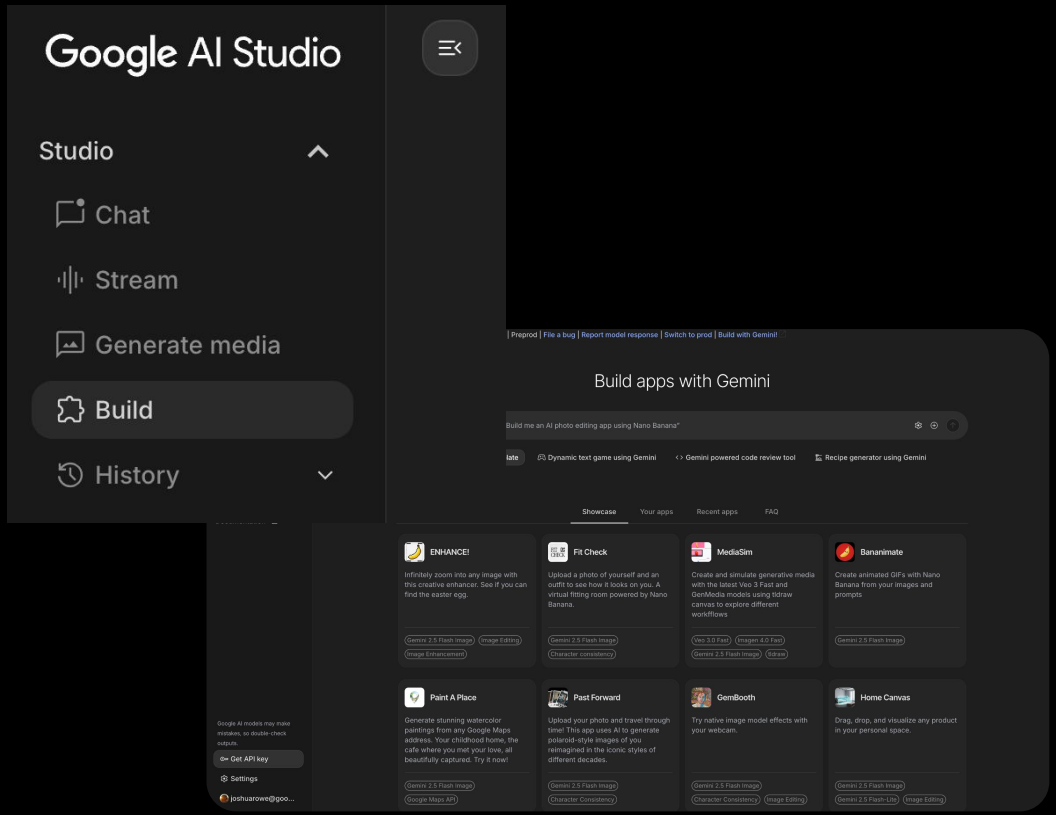
gemini.google.com



AI Studio - Build

Hosted development environment with fully managed development environment and deployment

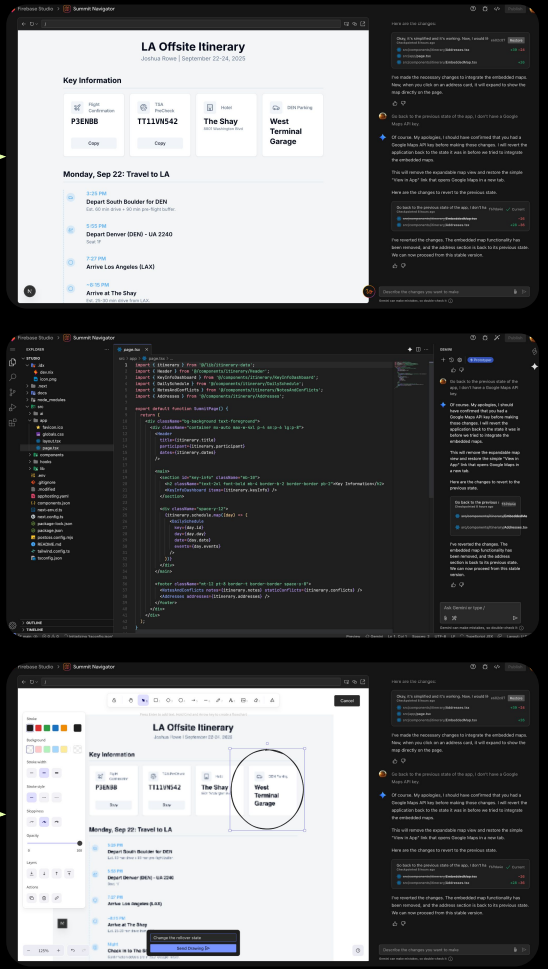
→ Tight integration with Gemini API
→ Full IDE
→ Set your own system prompt
→ Optimized for Angular or React



Firebase Studio

Hosted IDE with fully managed development environment and deployment

→ Prototyper mode - agent + live preview
→ Code mode - full IDE (VS Code)
→ Visual annotation and editing tools



Gemini CLI

Gemini in your Terminal

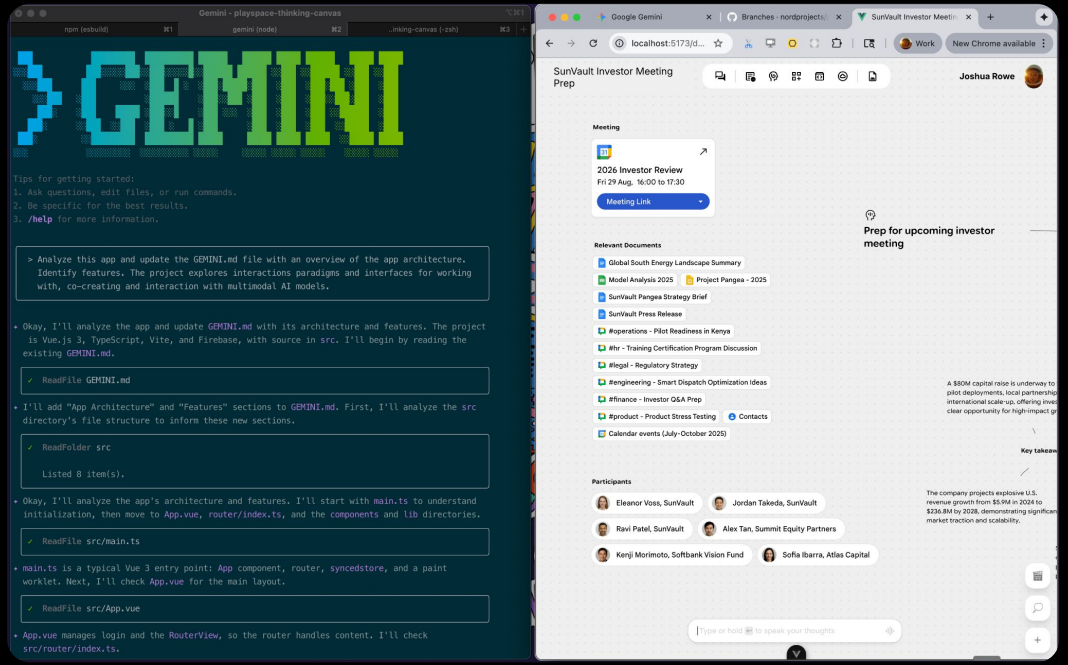
→ Local file system access
→ Execute all terminal commands
→ Manage local development environment
→ Natural language git* commands
g * the golden rule “commit early and often” learn git!
→ MCP servers

Install

```
npm install -g @google/gemini-cli
```

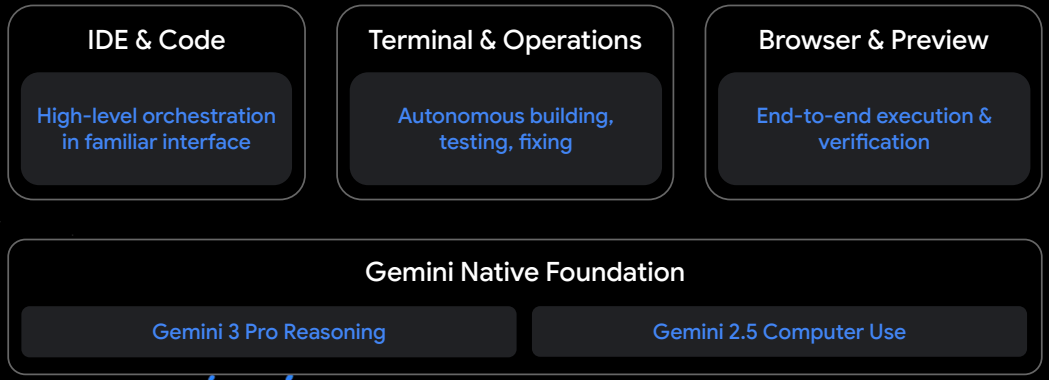
or, if you use brew:

```
brew install gemini-cli
```



Google Antigravity

Google Antigravity is an agentic development platform that empowers developers to act as architects, orchestrating autonomous agents that plan and execute complex tasks across the editor, terminal, and browser.



Proprietary & Confidential

Demo Time

PRIVILEGED & CONFIDENTIAL

Register for the Geotab Hackathon

From Idea to Prototyping

PRIVILEGED & CONFIDENTIAL

The industry is shifting from "vibe coding" to a structured approach—integrating AI with detailed specs, standards, and rigorous engineering fundamentals.

Vibe Coding Context Collapse

Relying solely on AI's "feel" for complex projects creates fragile systems. As the "big-picture" context fades, projects suffer from architectural decay and mounting technical debt.

Why the Collapse Happens

- **Loss of System Context:** AI excels in isolation but fails to maintain system-wide coherence, leading to broken dependencies.
- **The "Slop" Feedback Loop:** Low-quality AI output pollutes future training data, causing model degradation.
- **Developer Skill Gap:** Over-reliance creates a generation unable to debug deeply or understand underlying logic.

The Evolution: Prompt Engineering → Context Engineering

Context Engineering

System prompt, user prompt, attachments, tools, etc.

The art of providing the necessary context for an agent to accomplish a specific task

A system capable of autonomously performing a task.

From Visual Idea to a Working App

This phase is all about action. We'll take the concepts we brainstormed and turn them into a tangible, interactive prototype without writing complex code.

Start with your blueprint

Create Simple Mockups:
Use prompts like, "Create a simple UI layout for the app's home screen. It needs a logo at the top, a list of three items, and a navigation bar at the bottom." Gemini Canvas will generate a visual wireframe.

Define the User Journey:
Ideate with Gemini and and creates a clear flow for you to follow.

The Output:
A clear, visual plan

Build using AI Studio

Craft the Core Logic:
For example, if you're building a "Guide to AI," you'd use AI Studio to create and test the prompt that powers your jargon buster.

Get Your API Key:
This key is like a password that lets your final app use Gemini API.

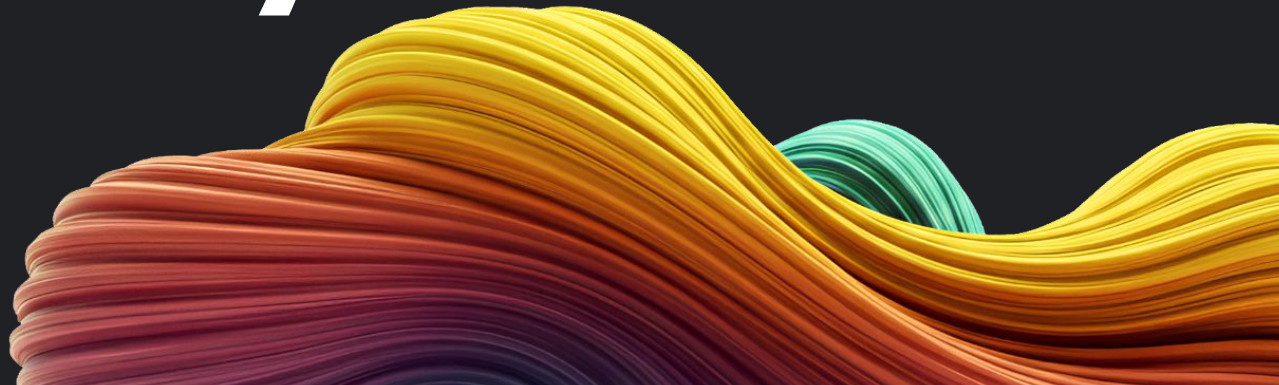
Deploy using Firebase Studio or Vertex AI

Deploy and Share:
With a single click, Firebase Studio builds the app and hosts it on a shareable URL. (You can use this link to submit for the hackathon)

For Prod grade deployments, you can use use Vertex AI too.

Proprietary

"Don't be a code monkey. Be the Architect. Let Antigravity be the contractor."

— The Future of Software

Thank you