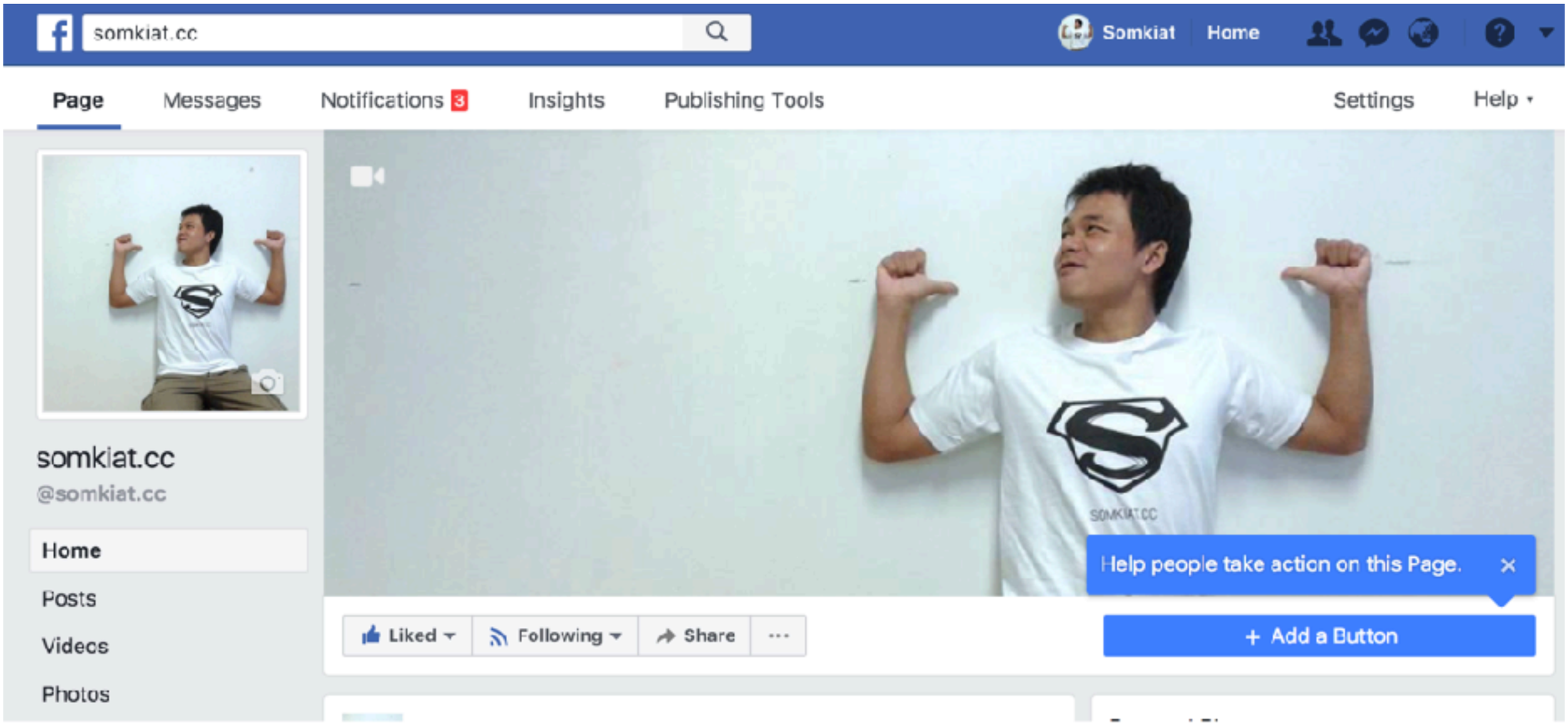
สยามชำนาญกิจ
SIAM CHAMNAN KIT

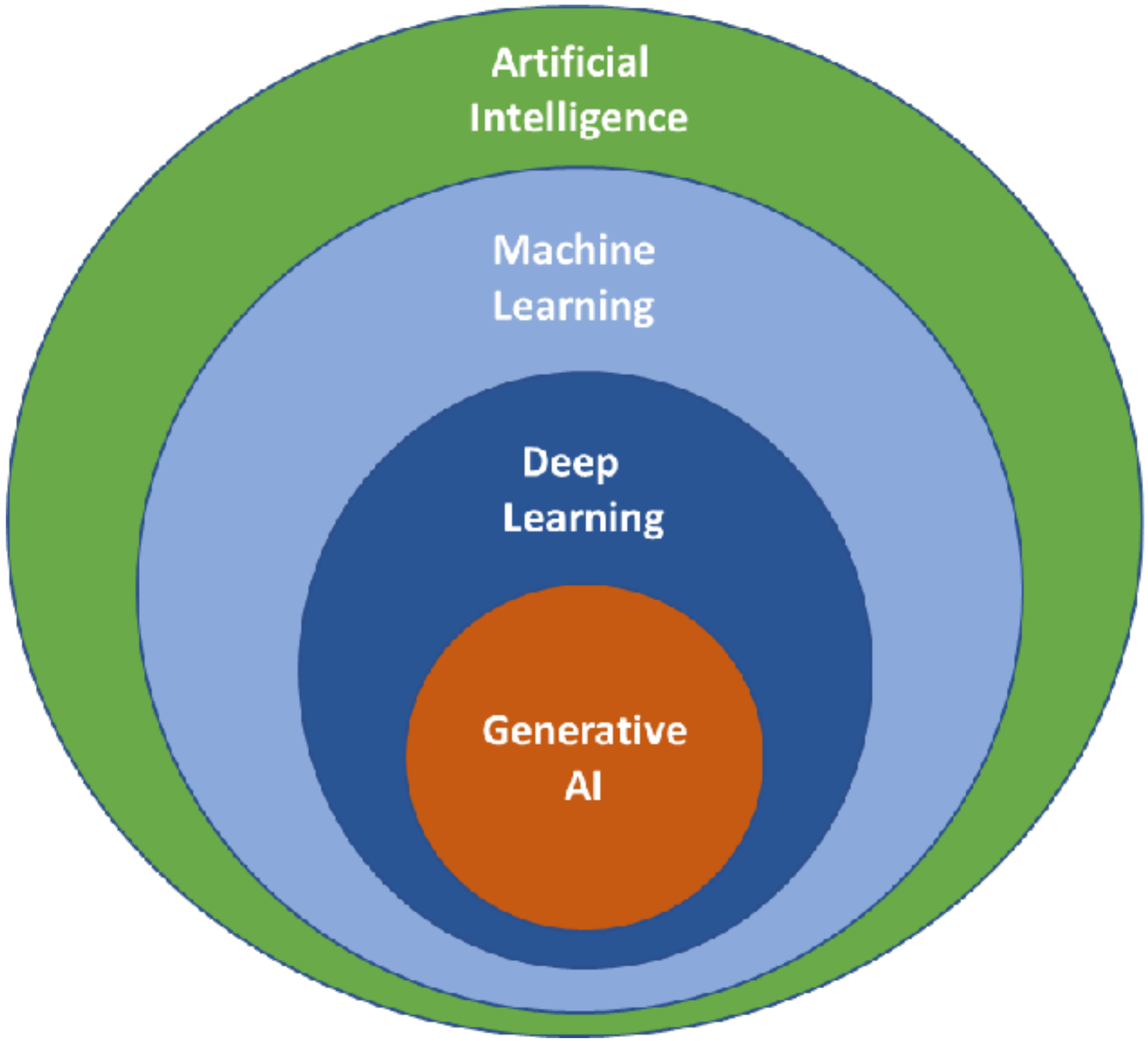
AI for
Software Development
Software Delivery



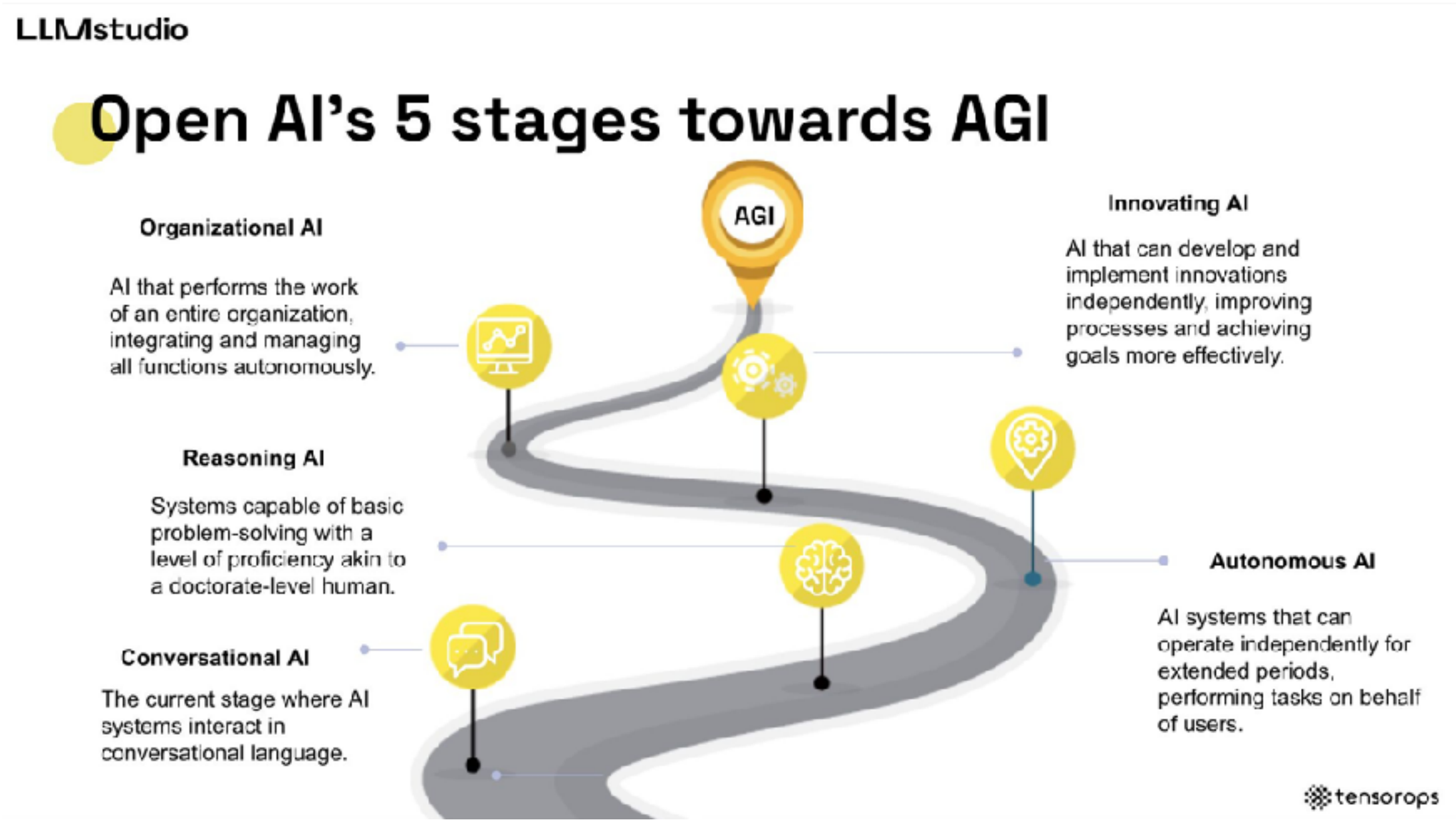
https://github.com/up1/workshop-ai-with-technical-team

Goals

Integrate Generative AI in Development
Optimize code quality
Team up with AI on coding tasks
Develop innovative solutions

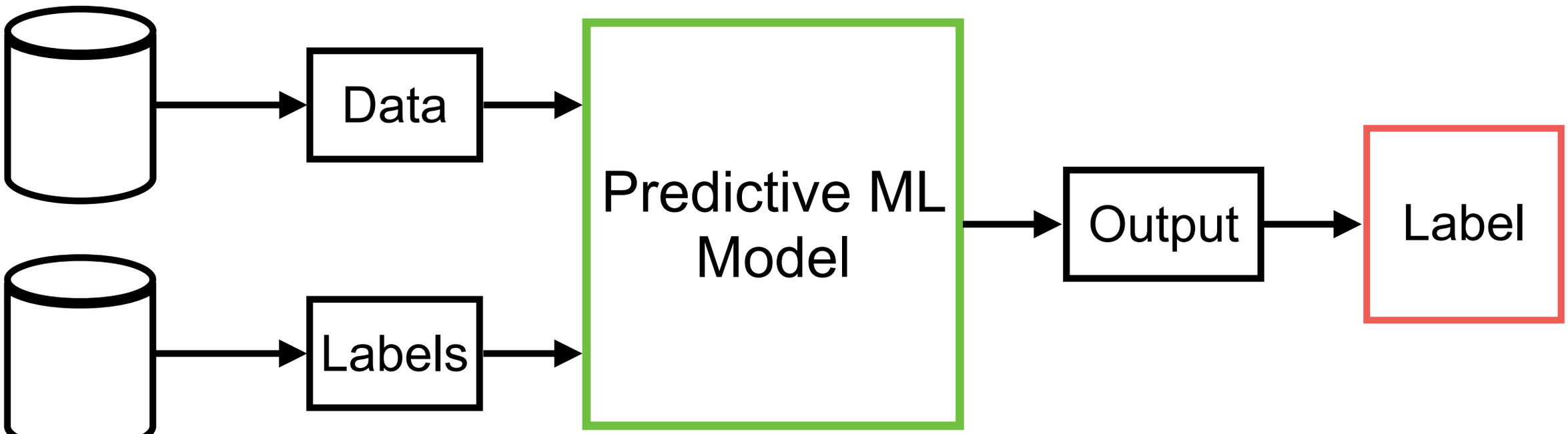


Stages of AI



https://www.tensorops.ai/post/openai-unveils-o1-model-the-biggest-leap-towards-agi-since-chatgpt

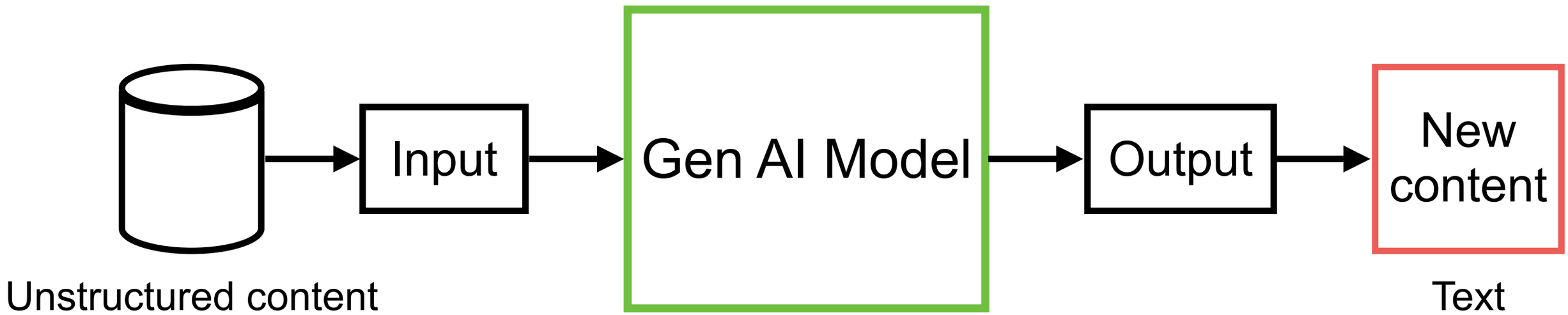
Machine Learning



Learn relationship
between data and label

https://grow.google/ai-essentials/

Generative AI

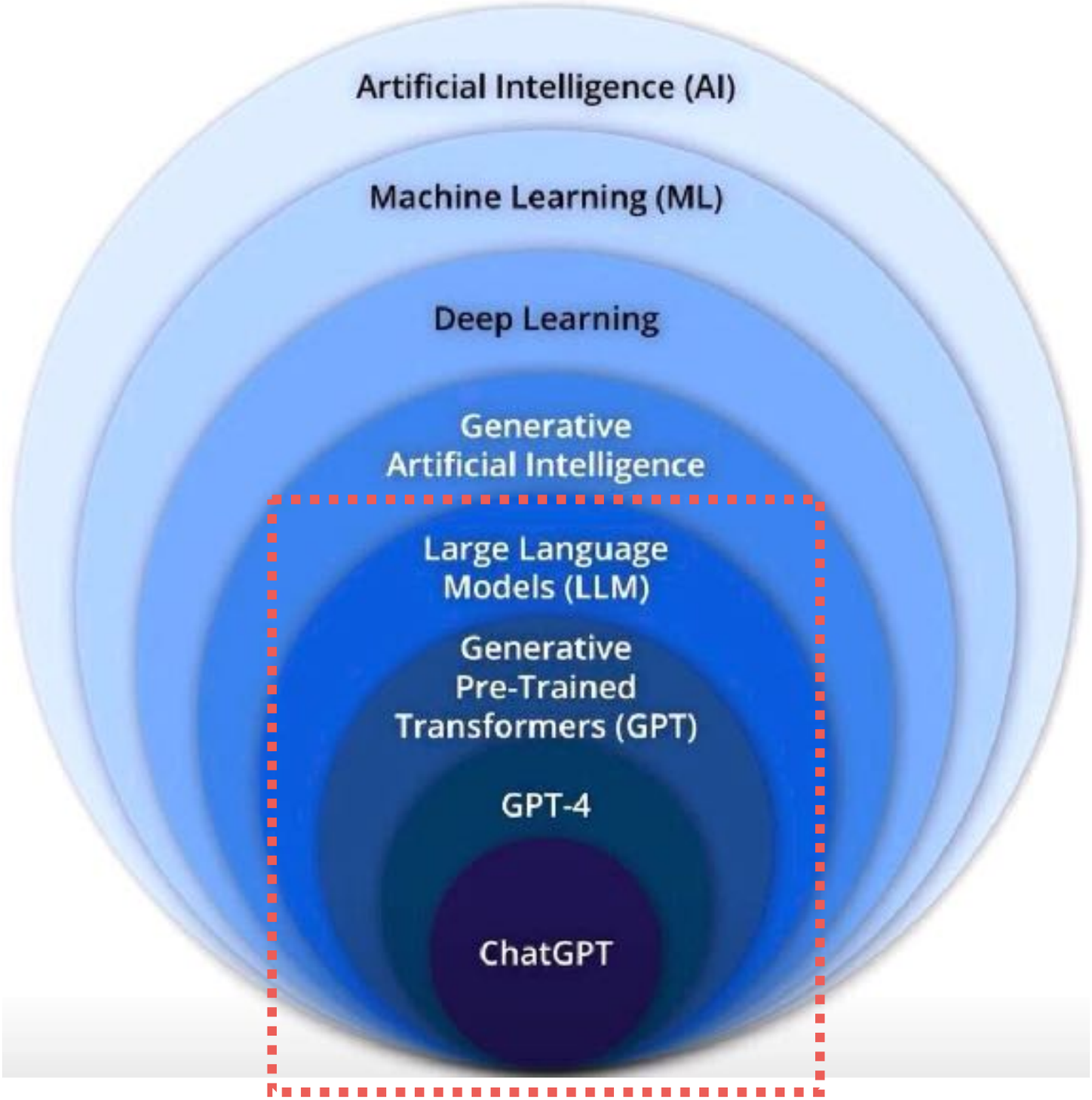


Unstructured content

Learn patterns from Unstructured content

Text
Image
Video
3D
Task

https://grow.google/ai-essentials/



Large Language Model (LLM)

Type of AI model
Process, understand
Generate human readable data

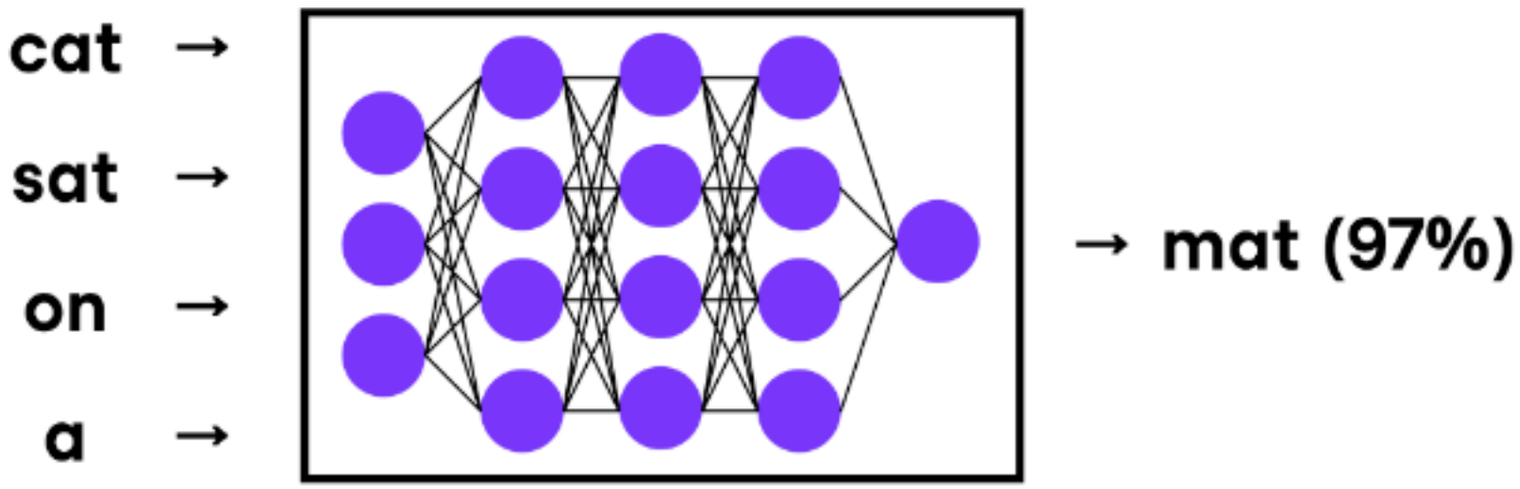
LLM

Training data !!

Large Language Model (LLM)

Neural network

Predicts the next word in a sequence



Let’s go !!

สีเหลือง

มะม่วง

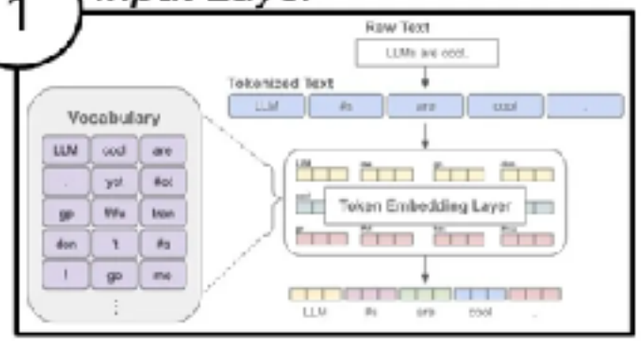
เตา

มีด

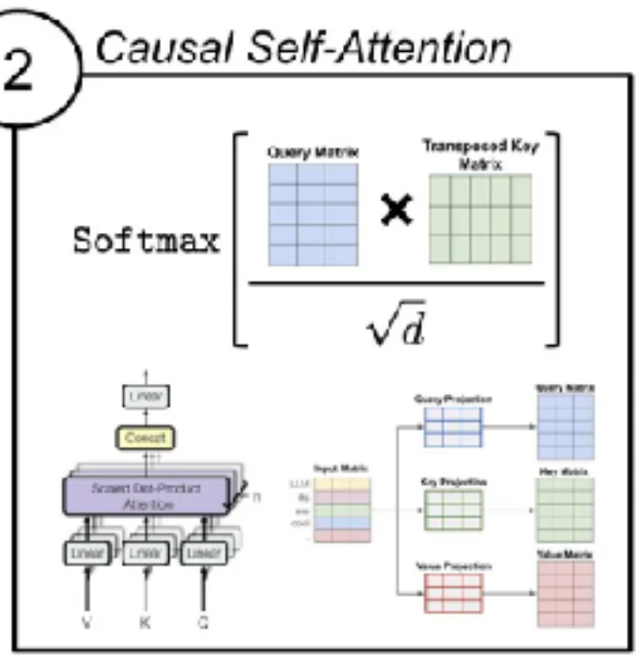
LLM components !!

Components of the Decoder-only Transformer

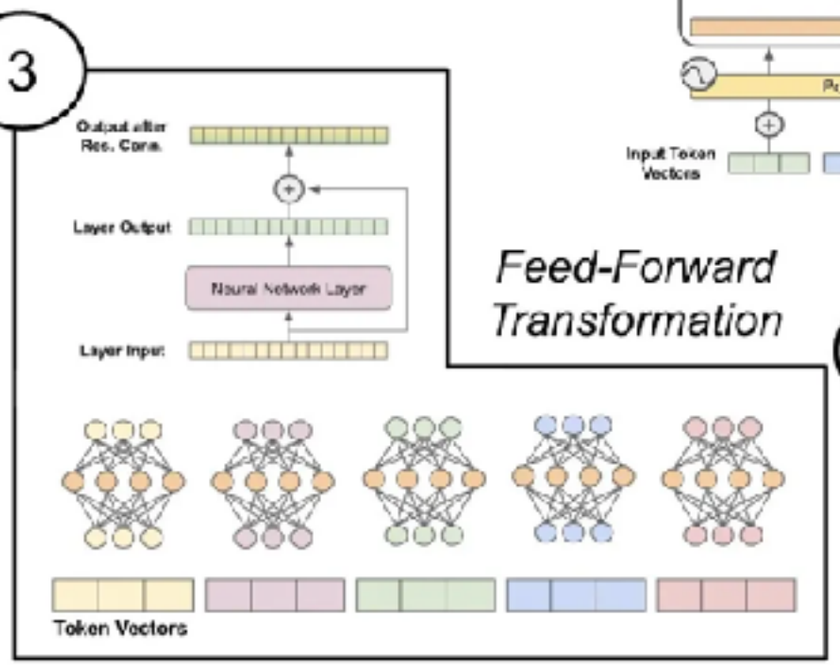
1 *Input Layer*



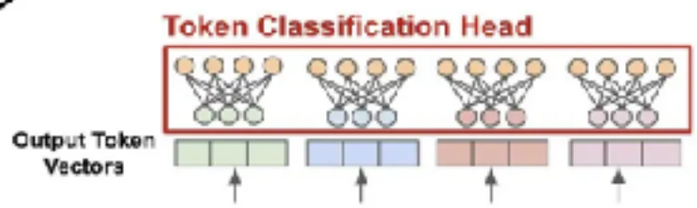
2 *Causal Self-Attention*

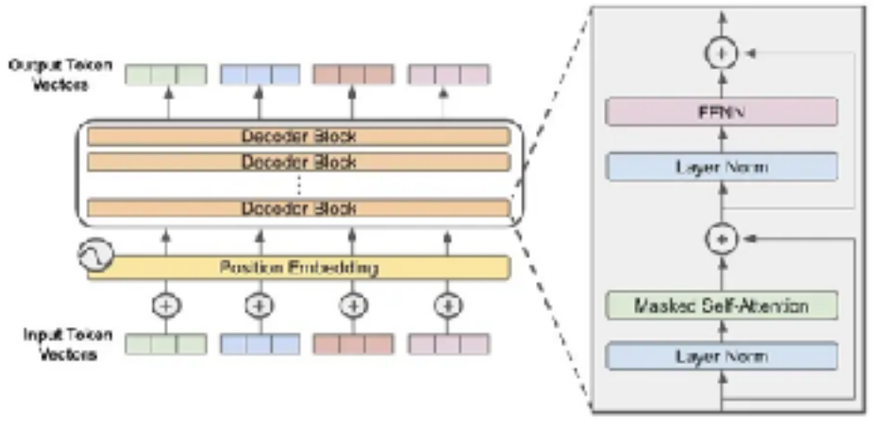


3 *Feed-Forward Transformation*



4 *Classification Head*

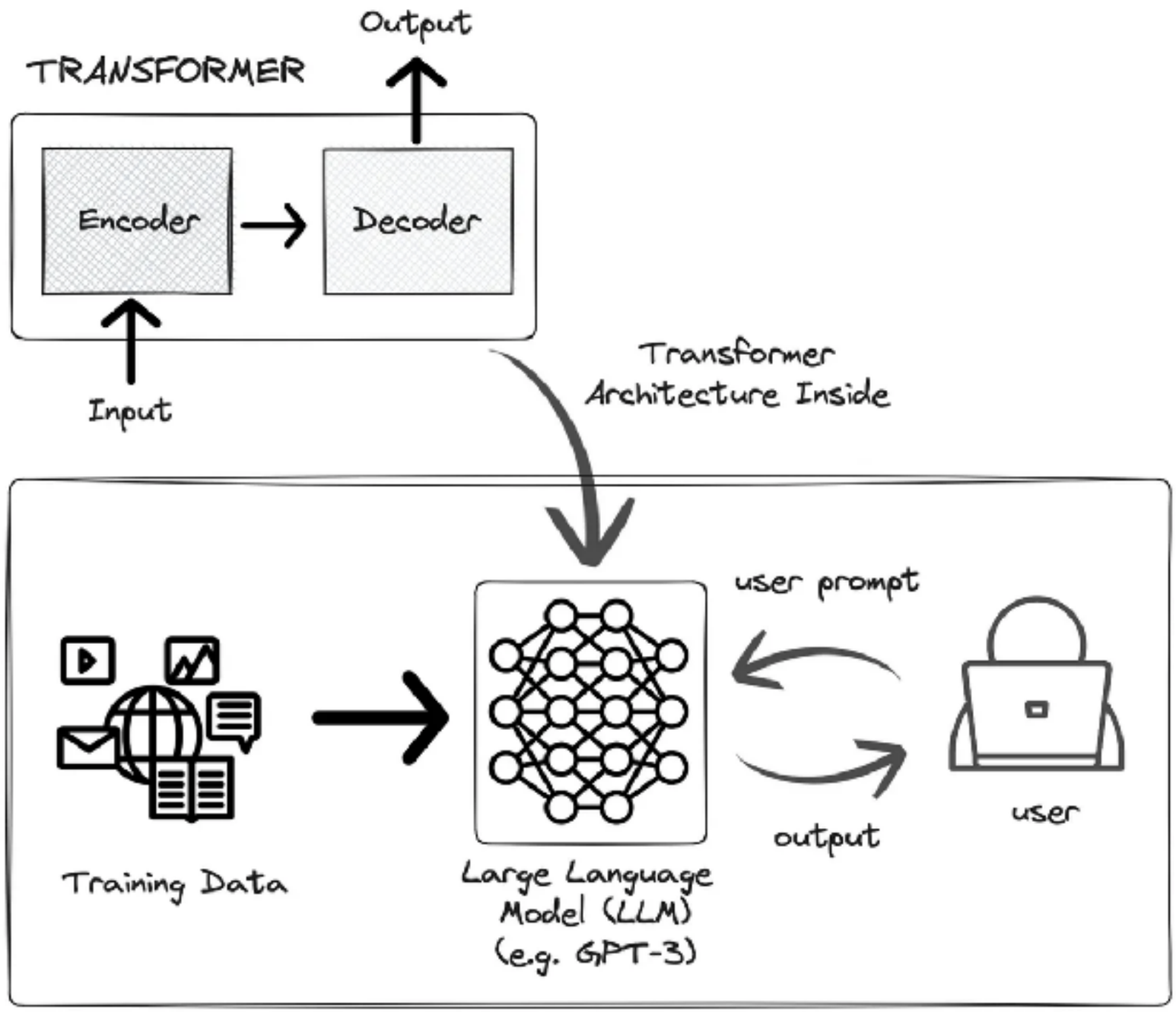




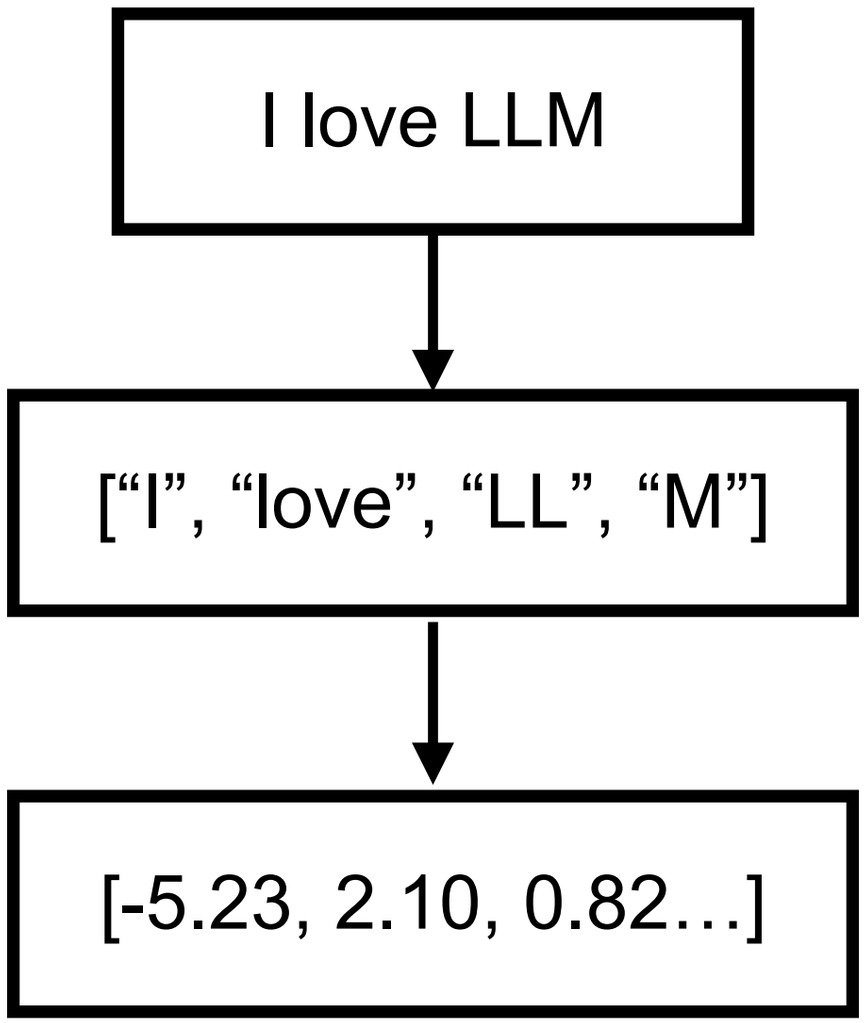
4 *Transformer Block*

https://stackoverflow.blog/2024/08/22/llms-evolve-quickly-their-underlying-architecture-not-so-much

Transformer inside



Transformer process

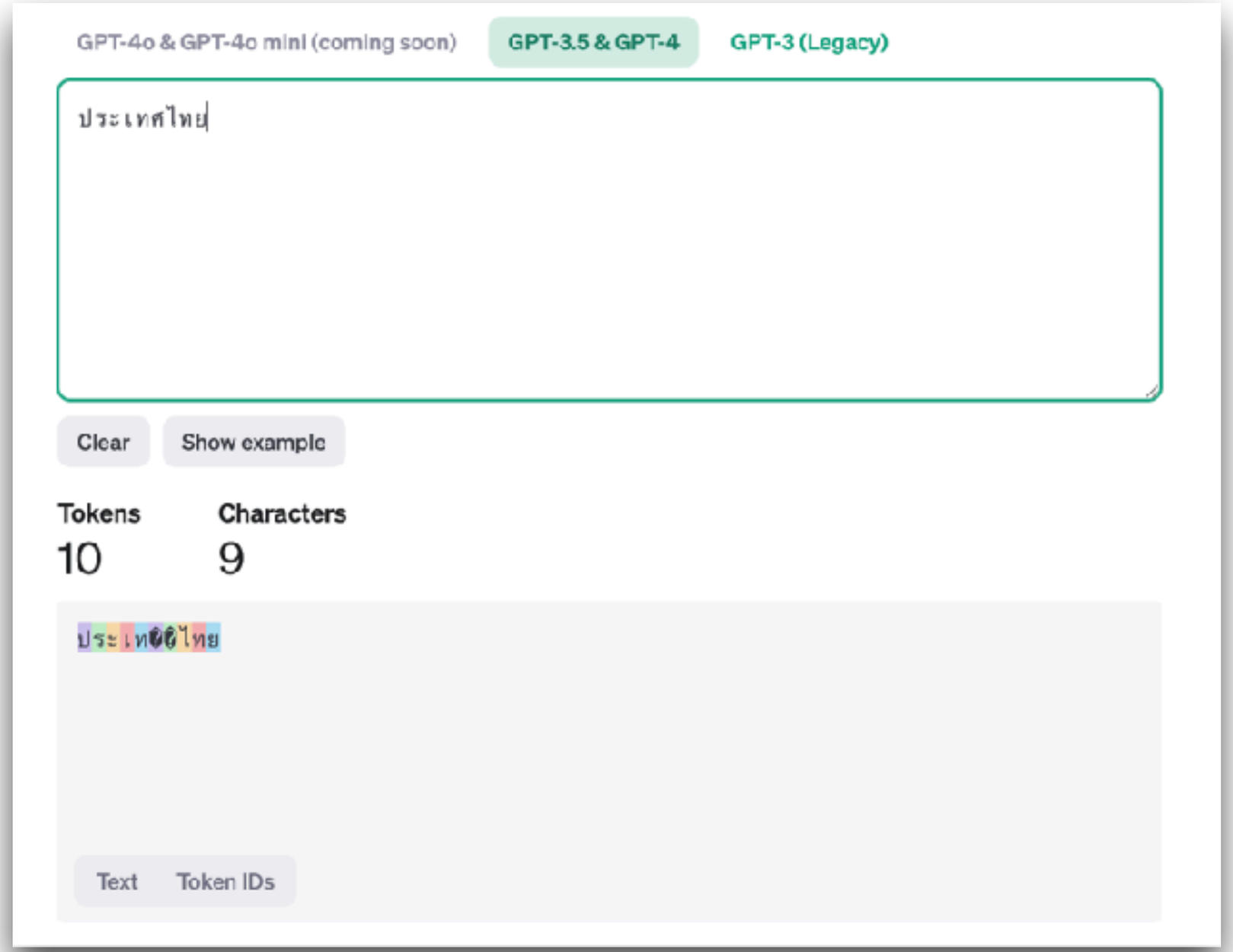


Sentence

Tokens

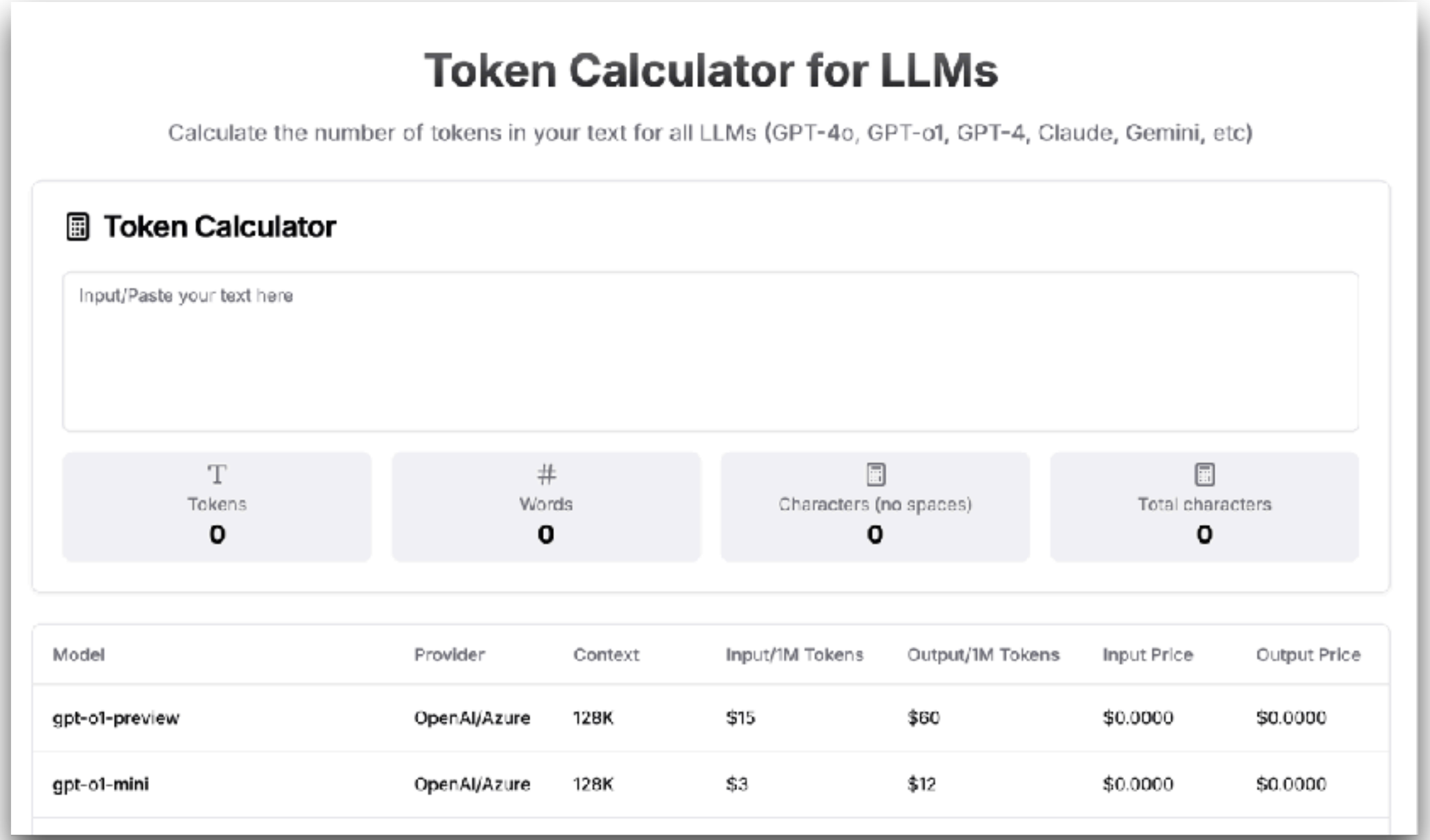
Embedding

OpenAI Tokenizer



https://platform.openai.com/tokenizer

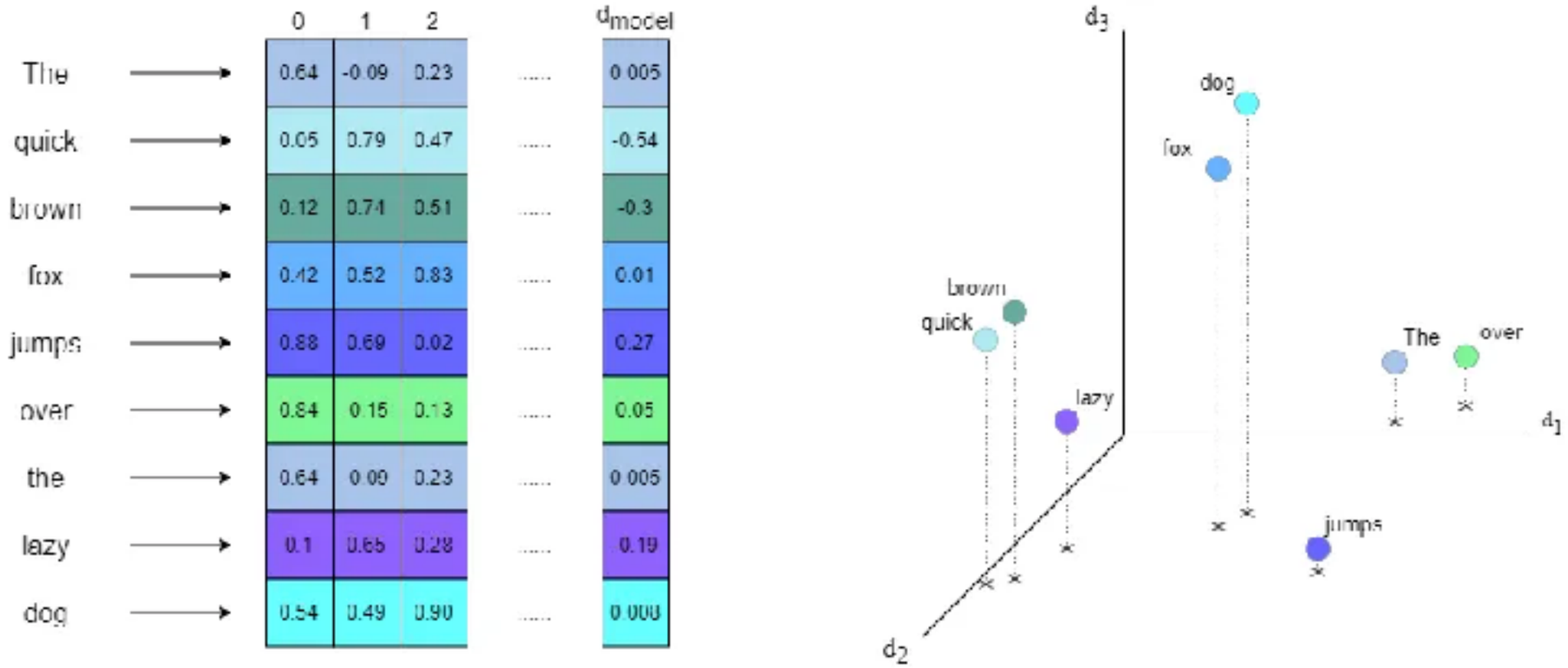
Token Calculator for LLM



https://token-calculator.net/

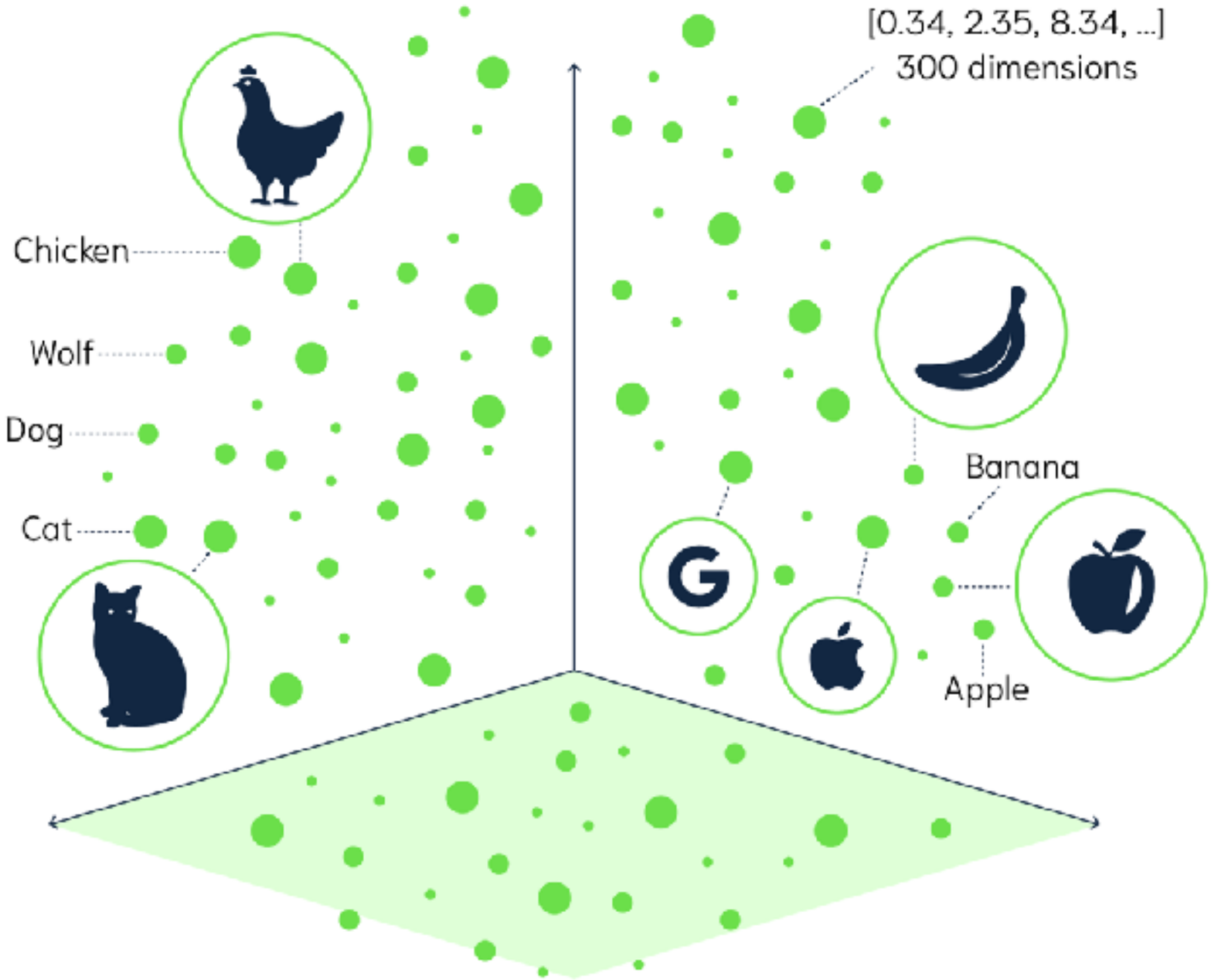
Embedding ?

Map items of **unstructured data** to high-dimensional real vectors

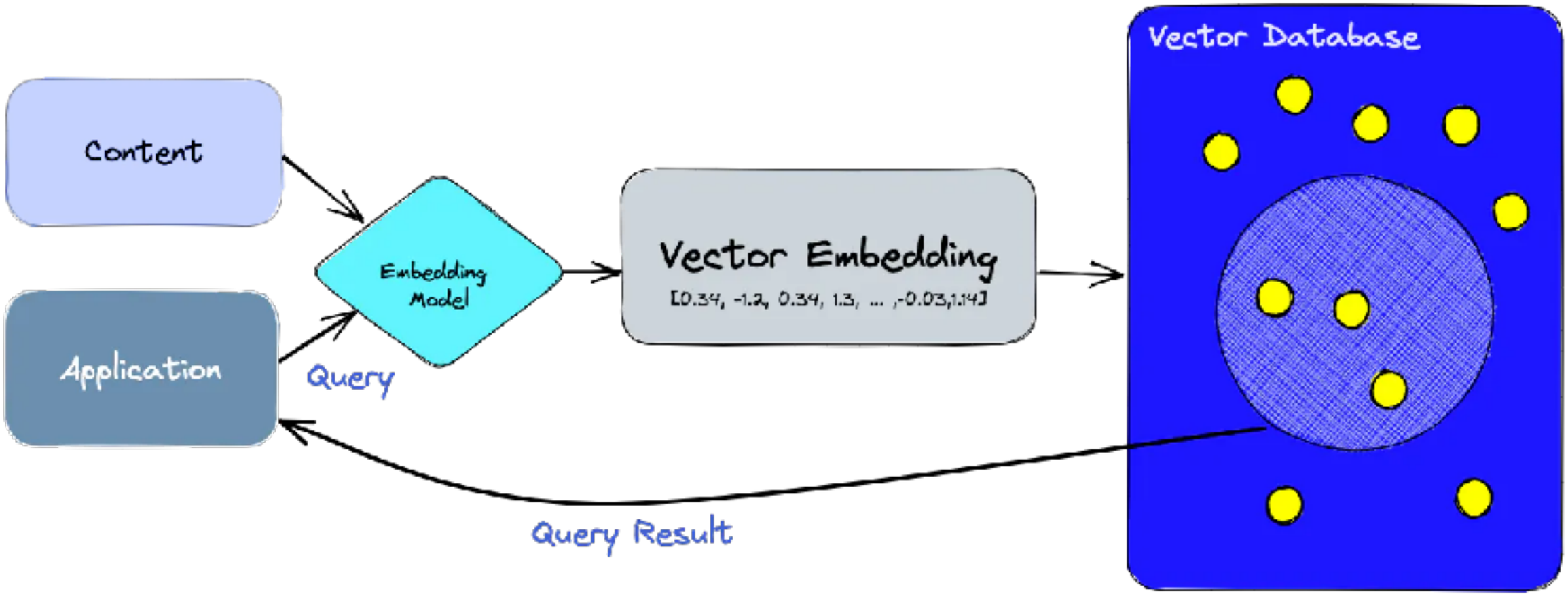


https://towardsdatascience.com/transformers-in-depth-part-1-introduction-to-transformer-models-in-5-minutes-ad25da6d3cca

Visual of Vector space

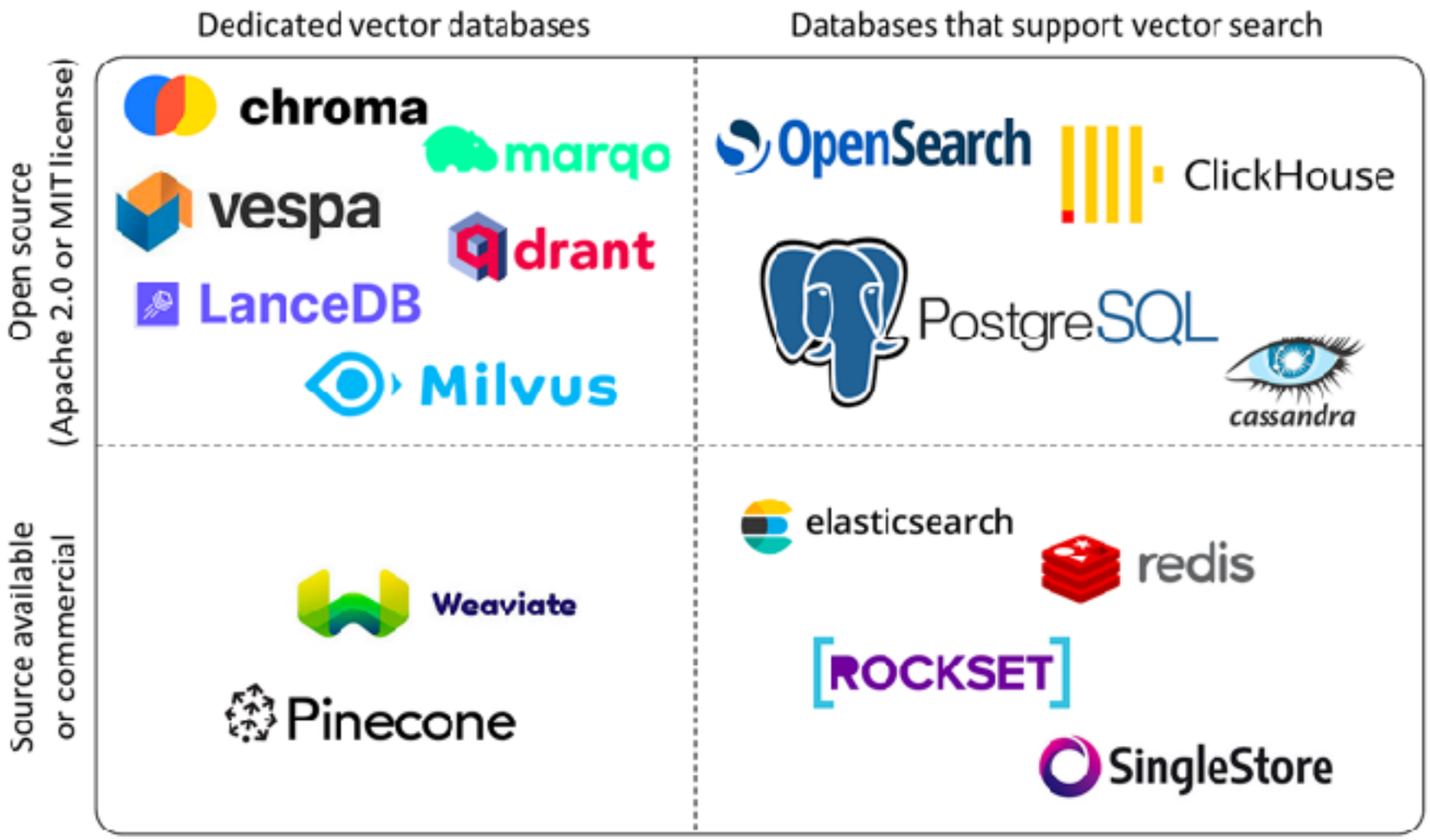


Store data in Vector Database



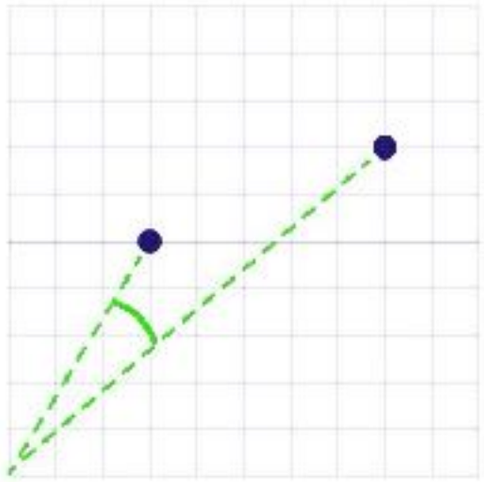
Vector Database ?

Map items of unstructured data to high-dimensional real **vectors**



https://towardsdatascience.com/transformers-in-depth-part-1-introduction-to-transformer-models-in-5-minutes-ad25da6d3cca

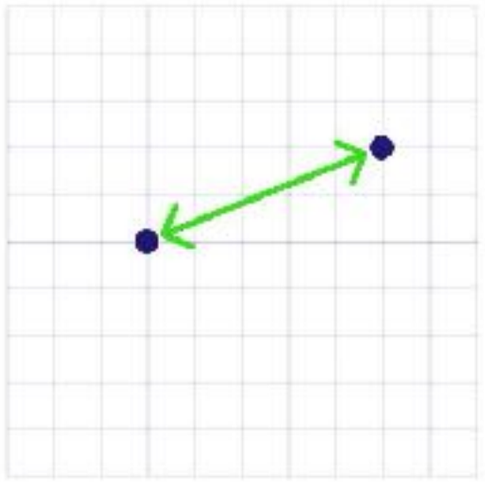
Distance Metrics in Vector Search



Cosine Distance

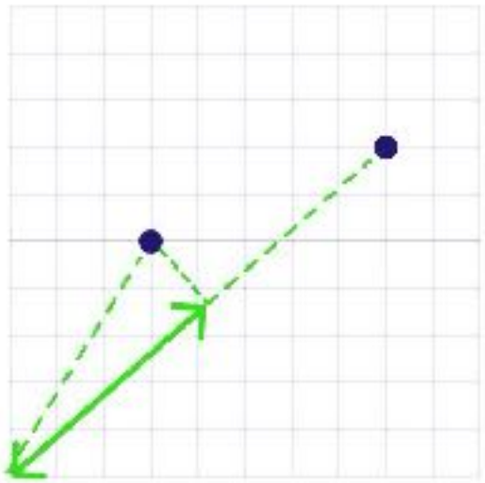
$$1 - \frac{A \cdot B}{||A|| \quad ||B||}$$

OpenAI



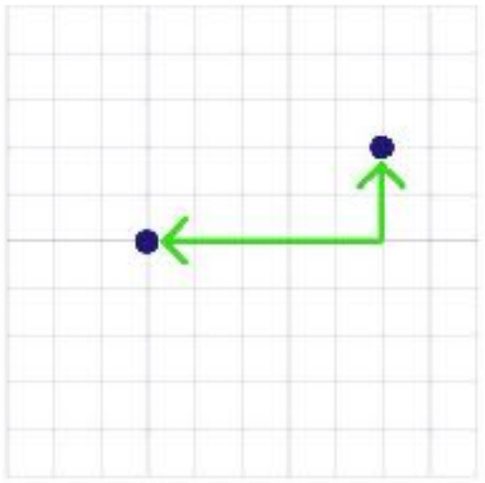
Squared Euclidean (L2 Squared)

$$\sum_{i=1}^{n} (x_i - y_i)^2$$



Dot Product

$$A \cdot B = \sum_{i=1}^{n} A_i B_i$$

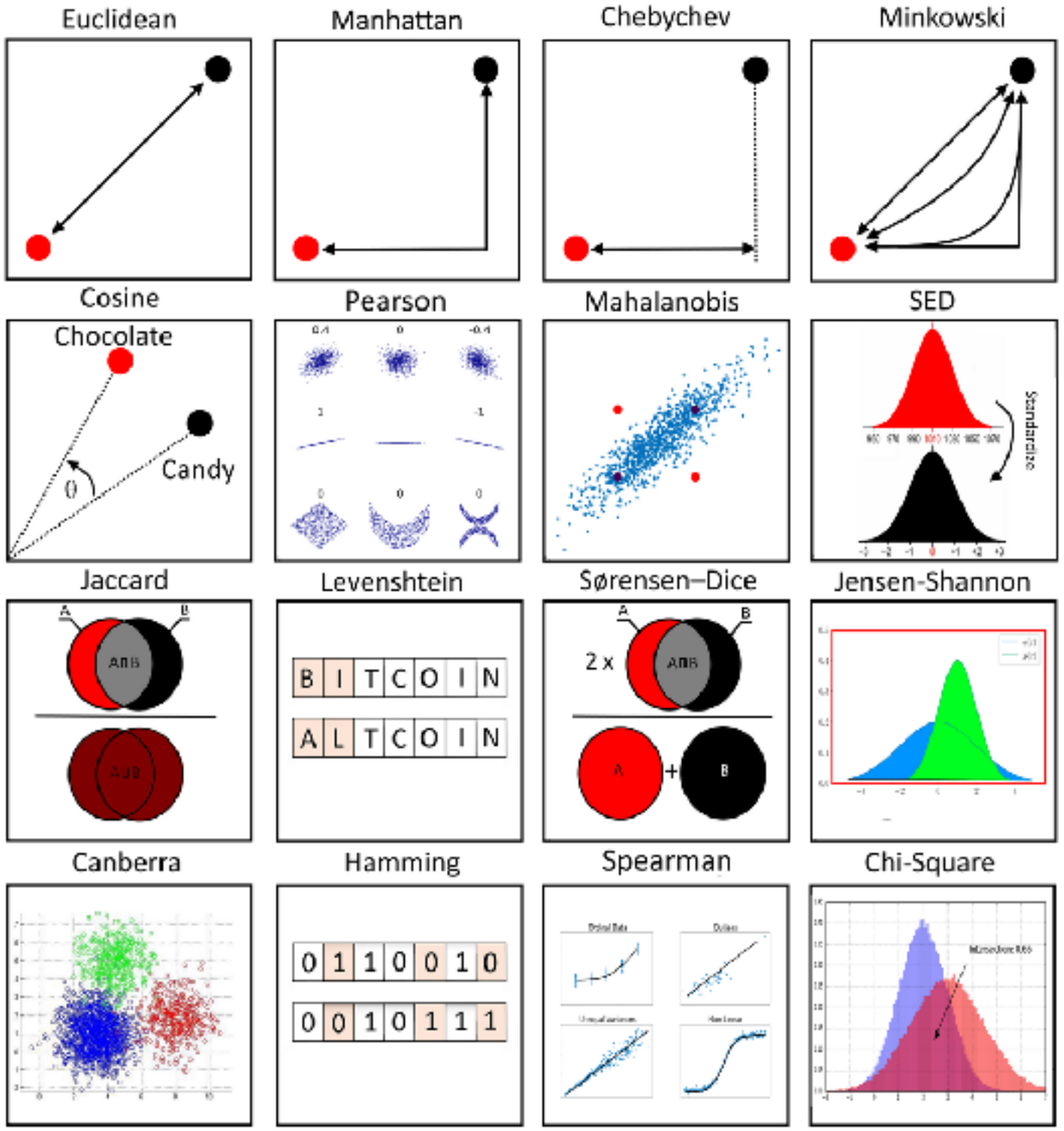


Manhattan (L1)

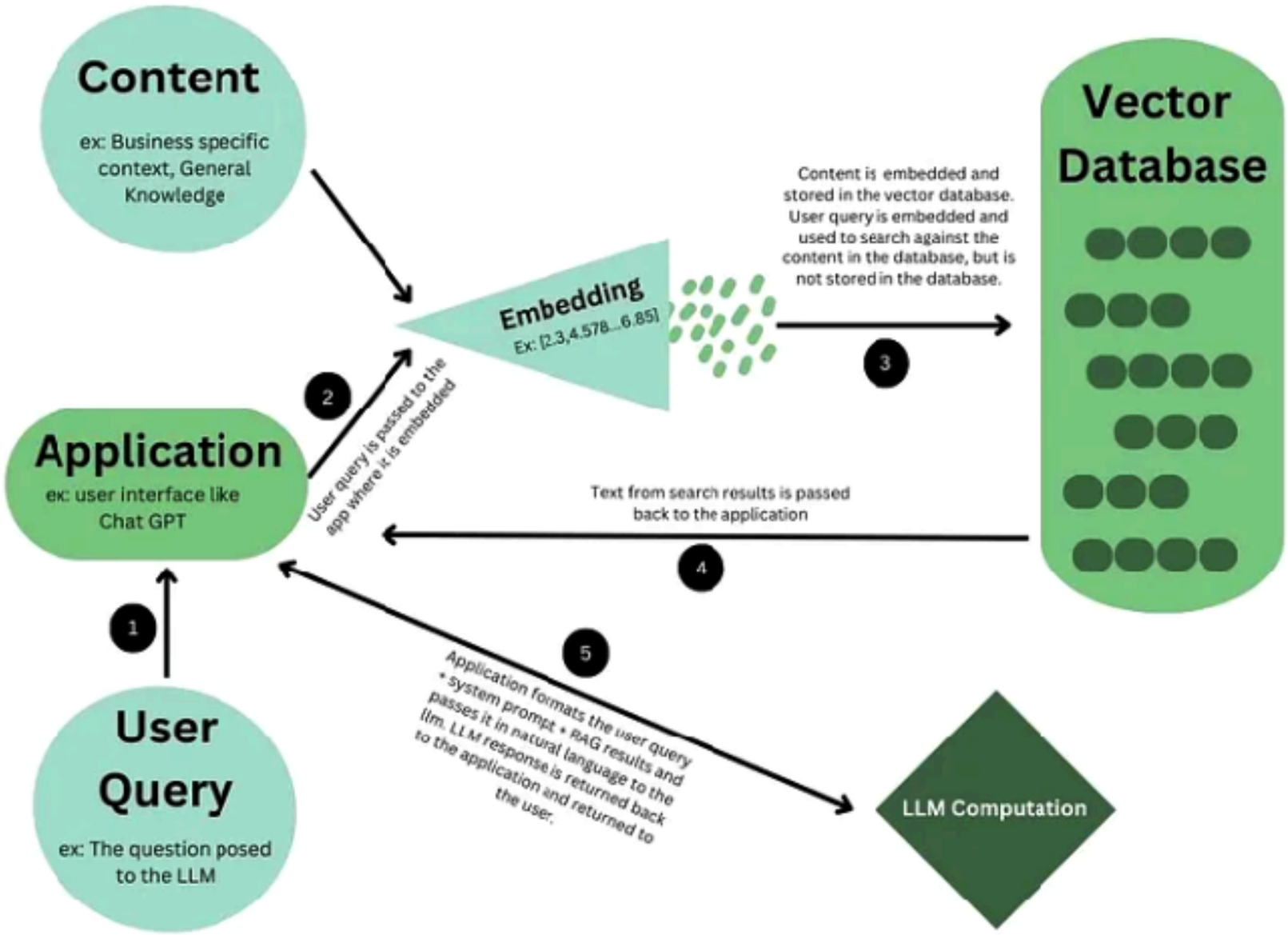
$$\sum_{i=1}^{n} |x_i - y_i|$$

https://help.openai.com/en/articles/8984345-which-distance-function-should-i-use

Distance measure in Data Science

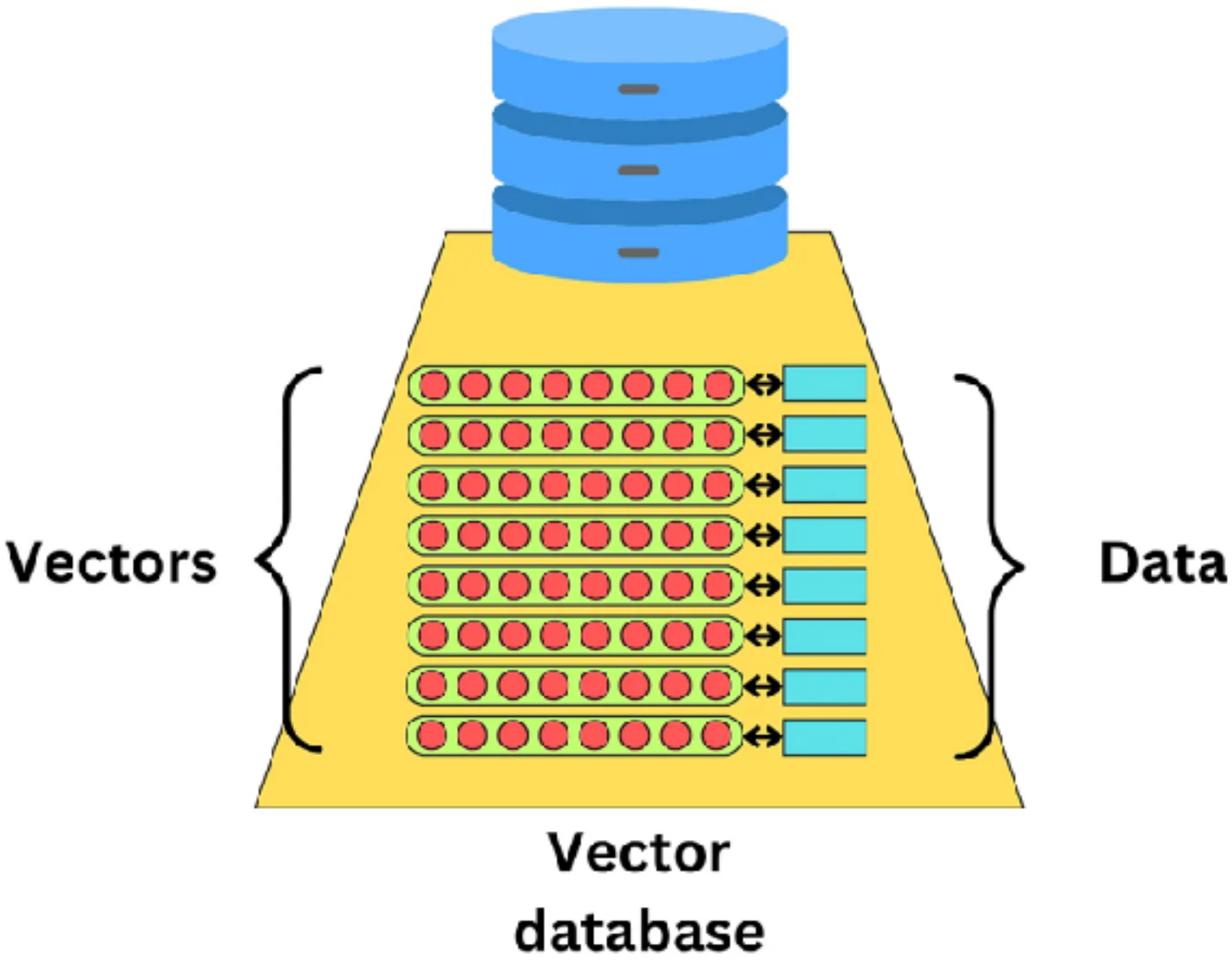


Basic Flow



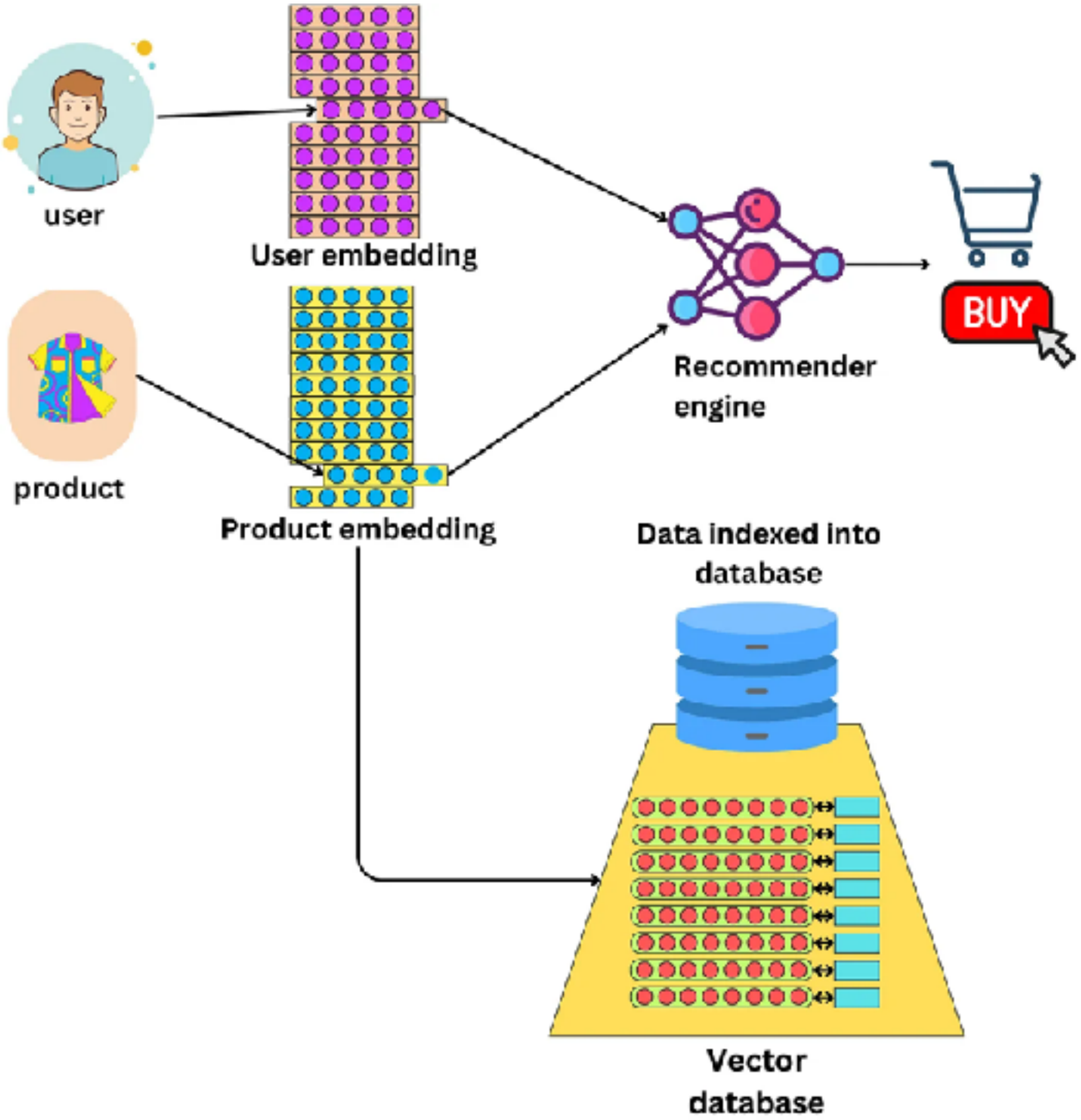
Vector Database

Indexing data with embedding



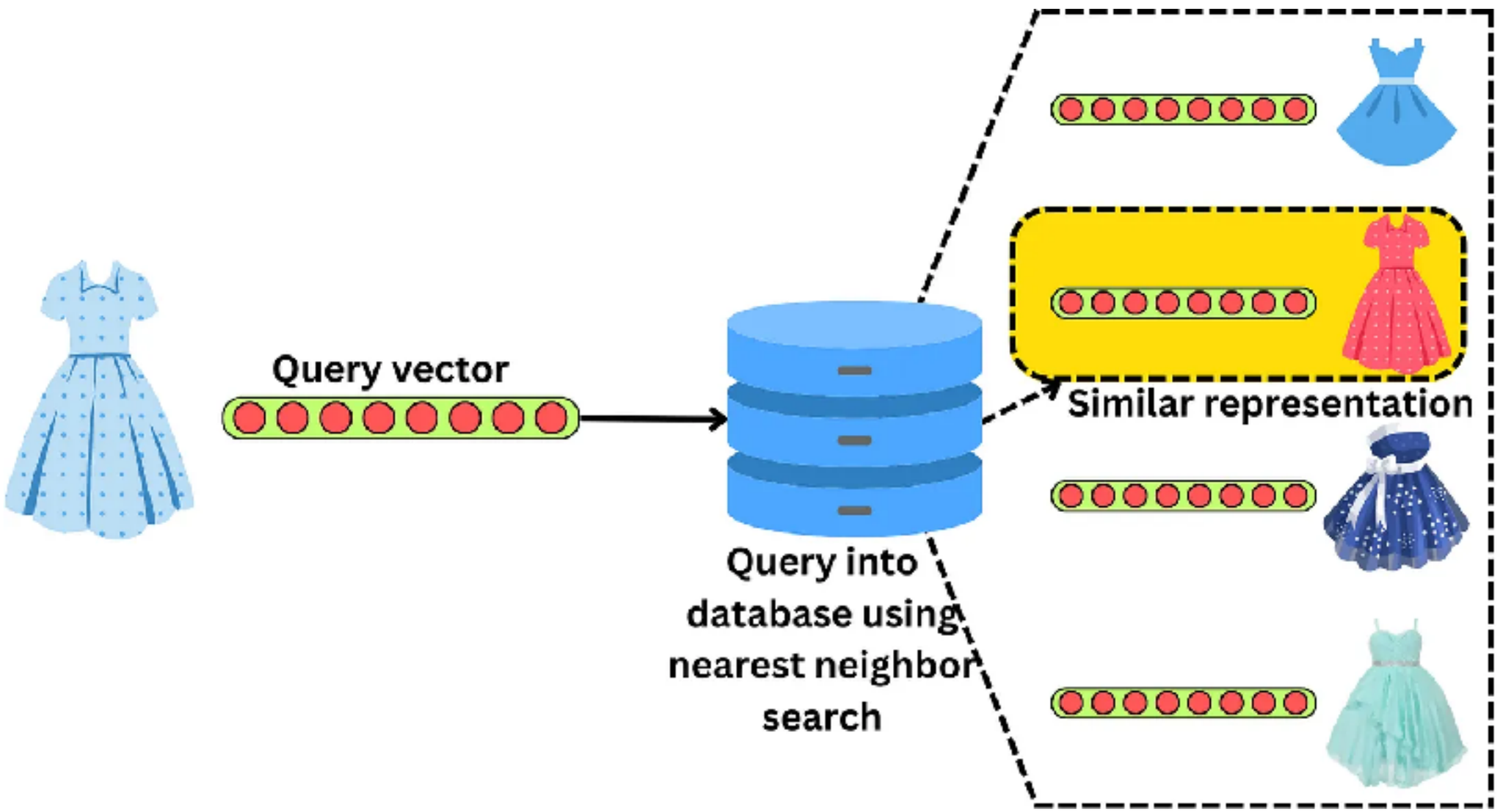
https://newsletter.theaiedge.io/p/understanding-how-vector-databases

Embedding data



https://newsletter.theaiedge.io/p/understanding-how-vector-databases

Search similar items



https://newsletter.theaiedge.io/p/understanding-how-vector-databases

LLMs in industry

Model name	Company
Bidirectional Encoder Representation from Transformers (BERT)	Google AI
Generative Pre-trained transformer-3 (GPT-3)	OpenAI
Generative Pre-trained transformer-4 (GPT-4)	OpenAI
Pathways Language Model-E (PaLM-E)	Google AI
BLOOM	NVIDIA AI
Llama 3	Facebook
Claude 3.5 Sonnet	Anthropic

LLM Development timeline

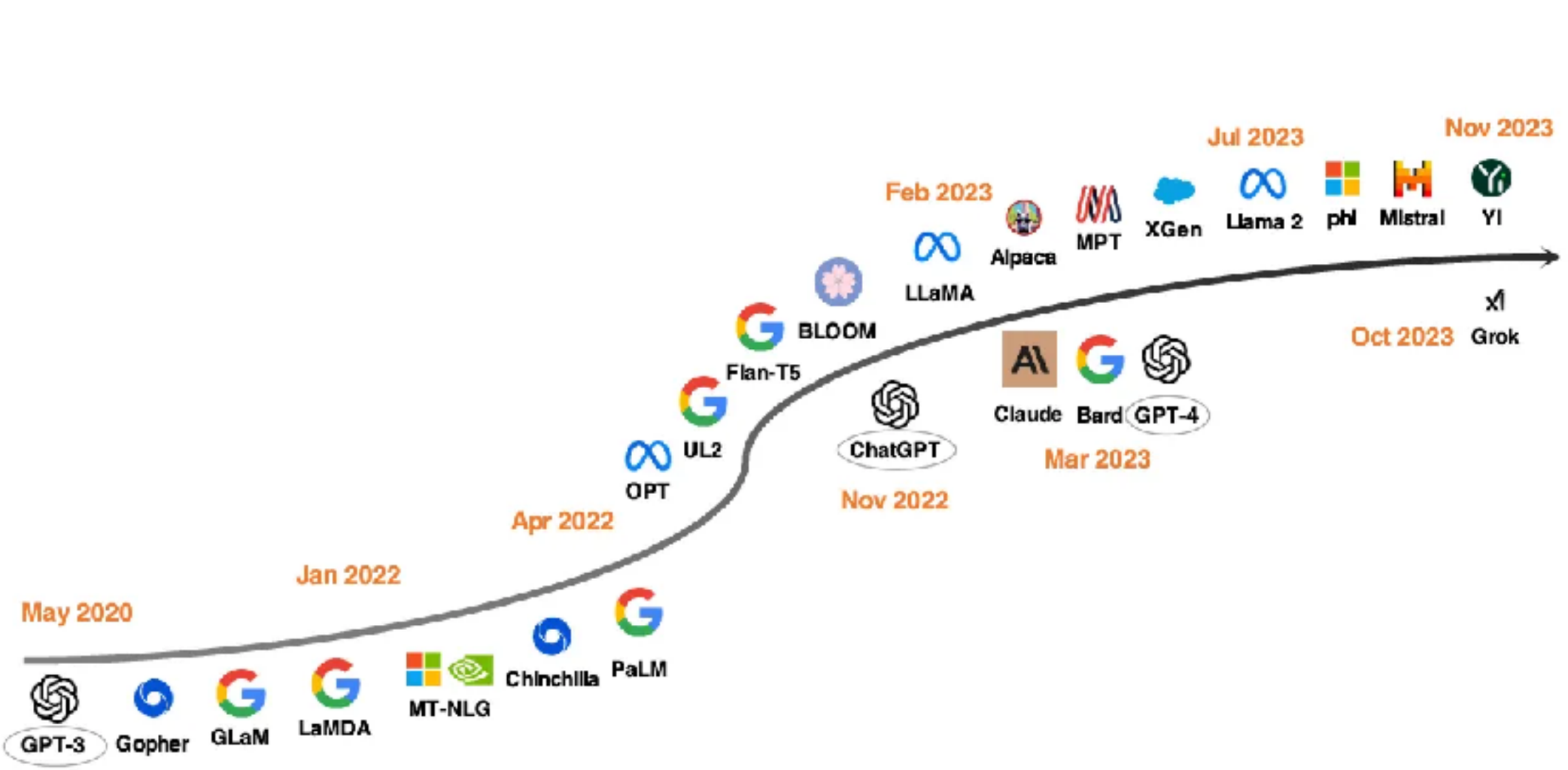
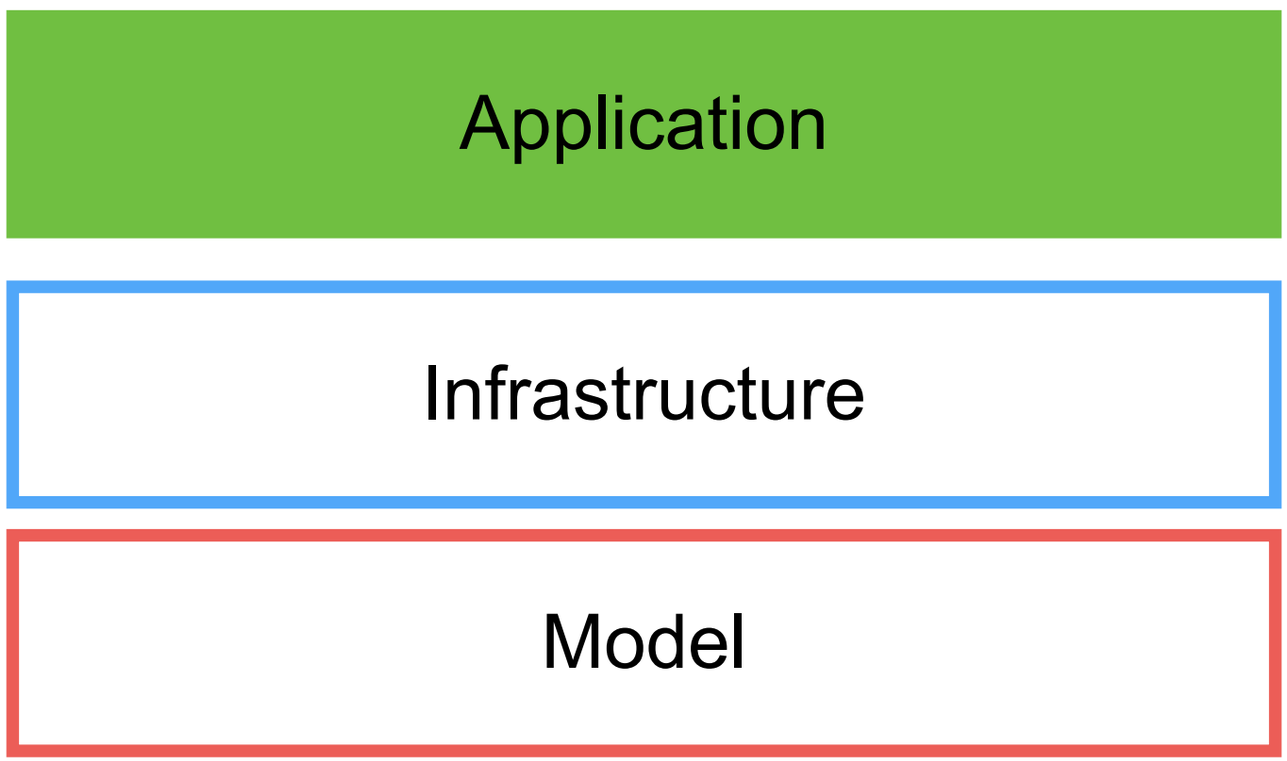
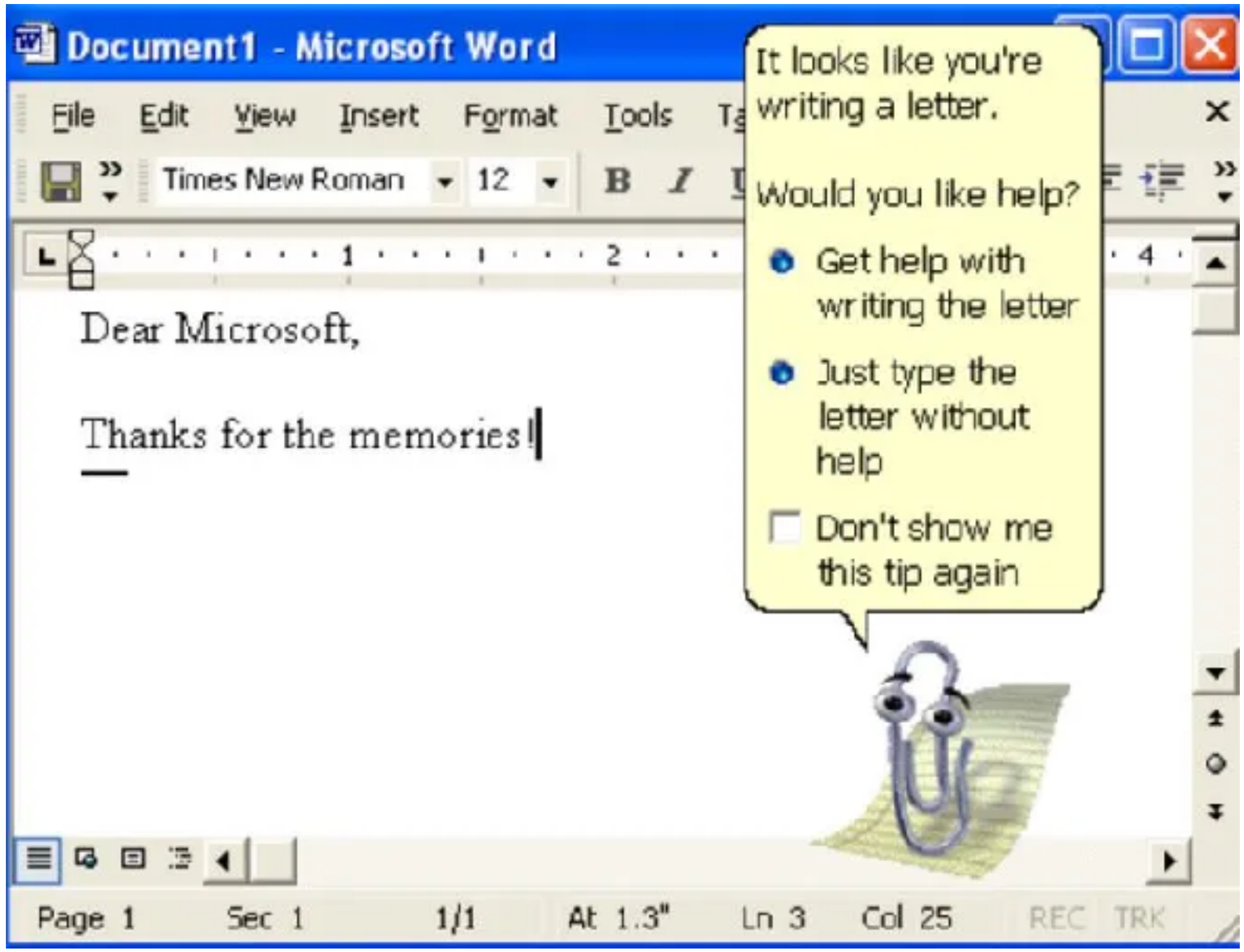


Figure 3: LLM development timeline. The models below the arrow are closed-source while those above the arrow are open-source.

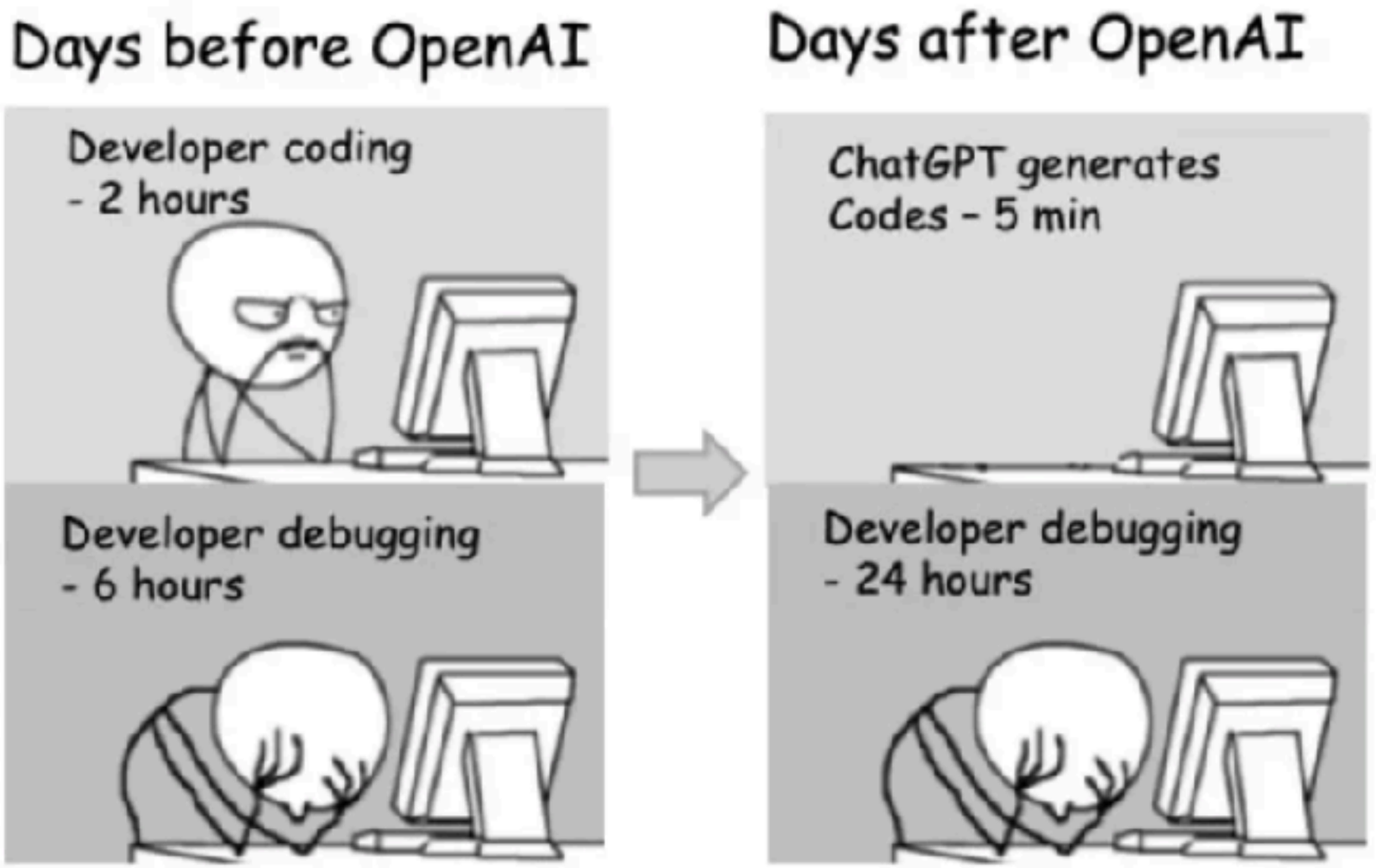
https://arxiv.org/abs/2311.16989



Generative AI (GenAI)



Generative AI (GenAI)

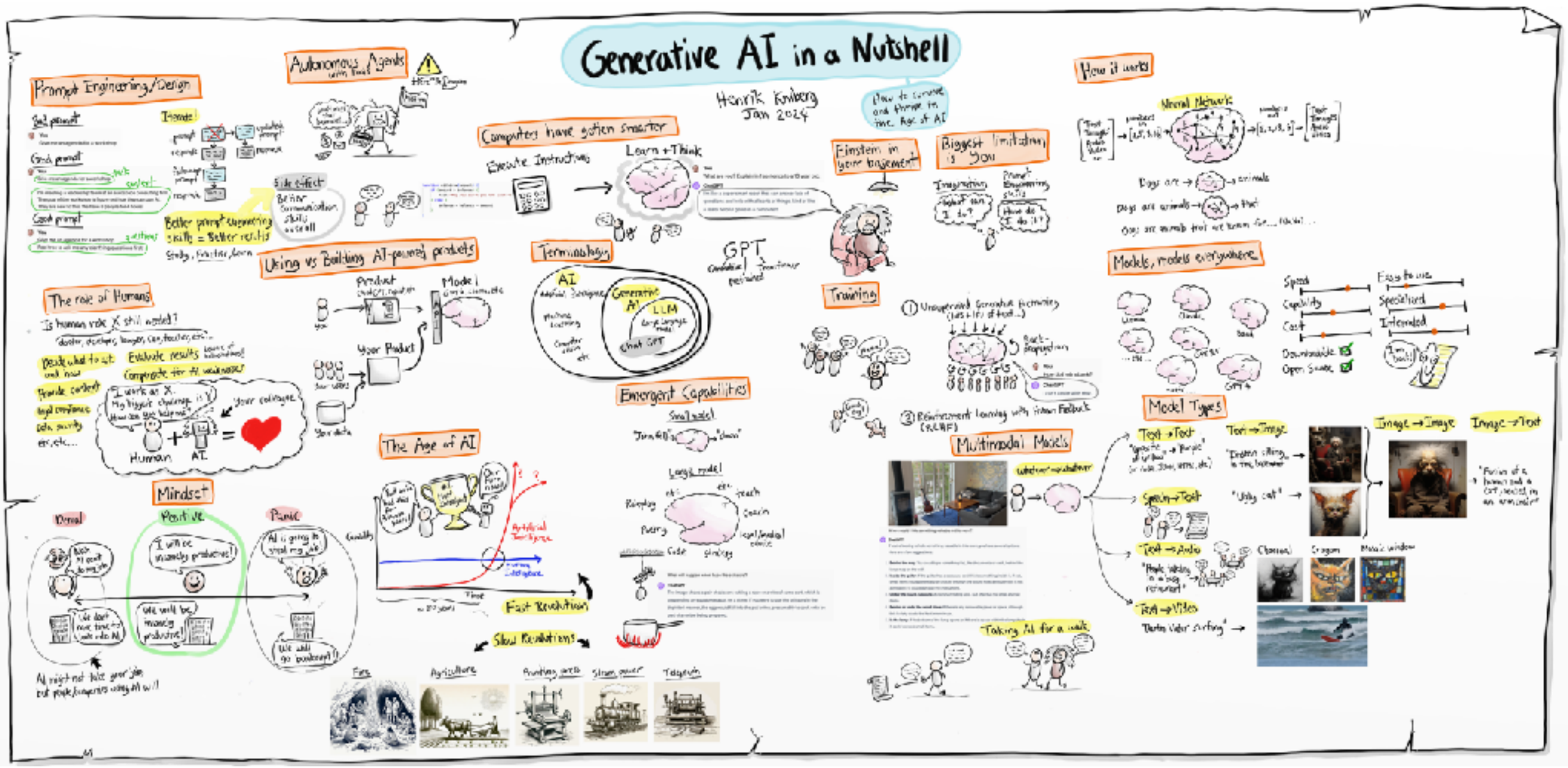


Trust, but verify output !!

Challenges in Gen AI

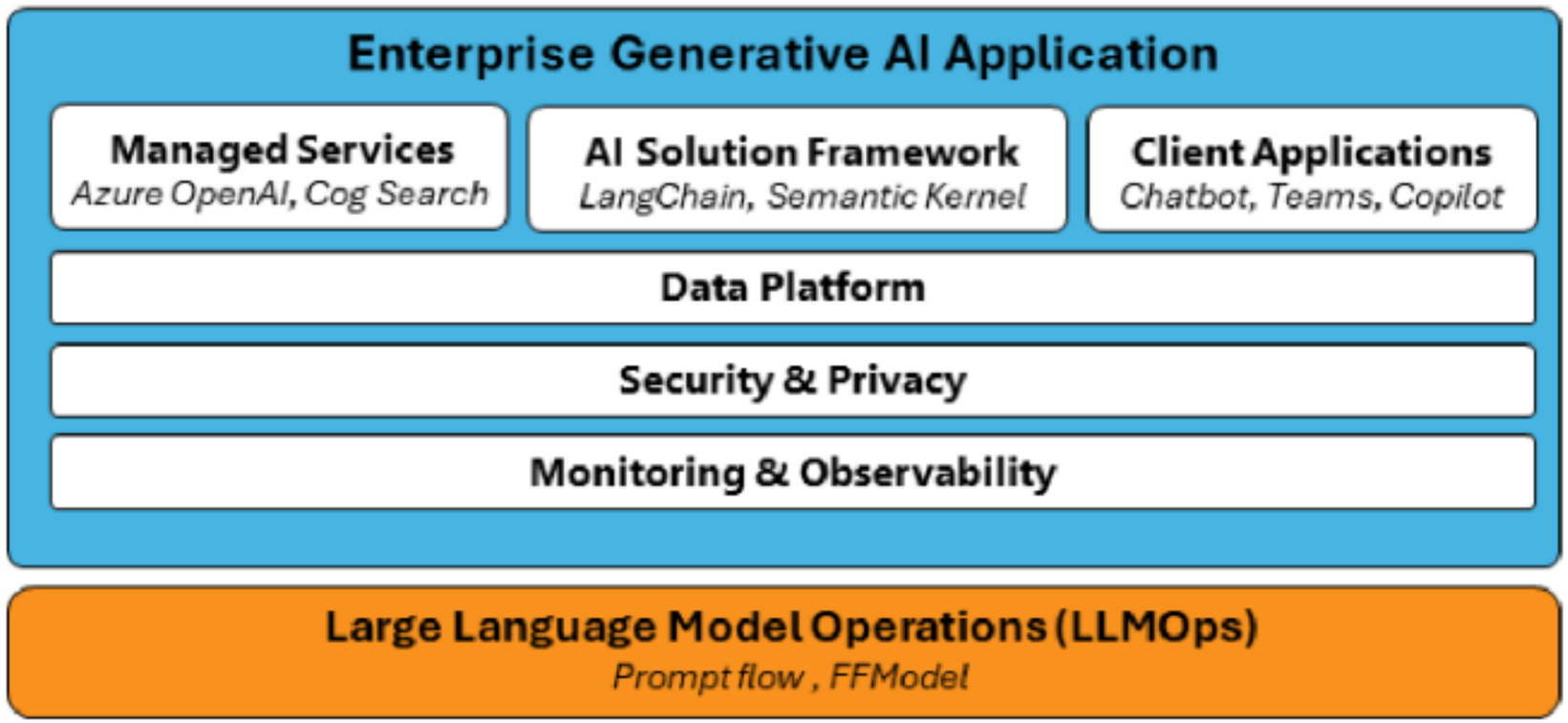
Lack of high-quality data
Data licenses
Generation latency
High computational power

Generative AI in Nutshell



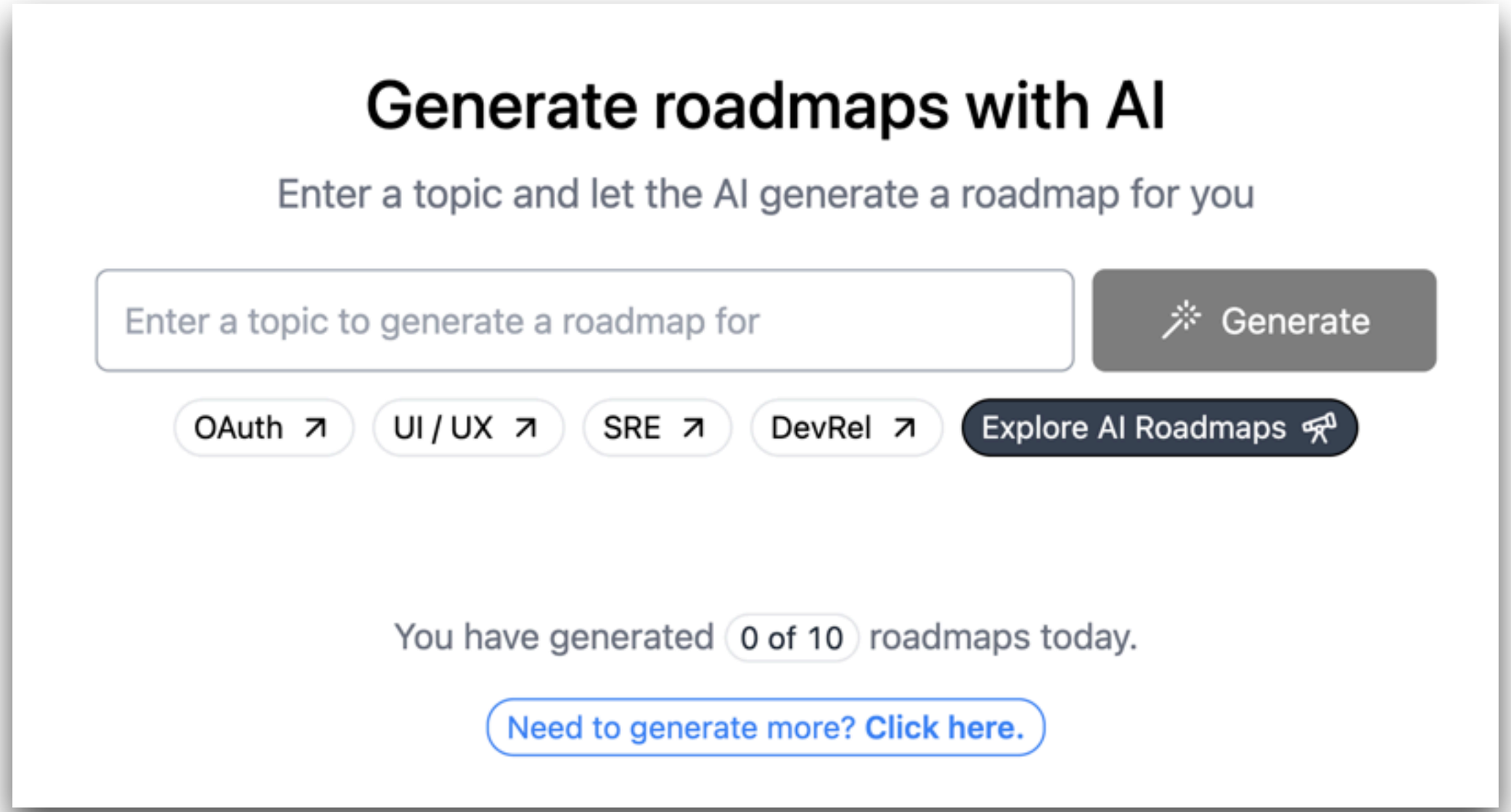
https://www.youtube.com/watch?v=2IK3DFHRFfw

Generative AI Application Stack



https://learn.microsoft.com/en-us/ai/playbook/technology-guidance/generative-ai/

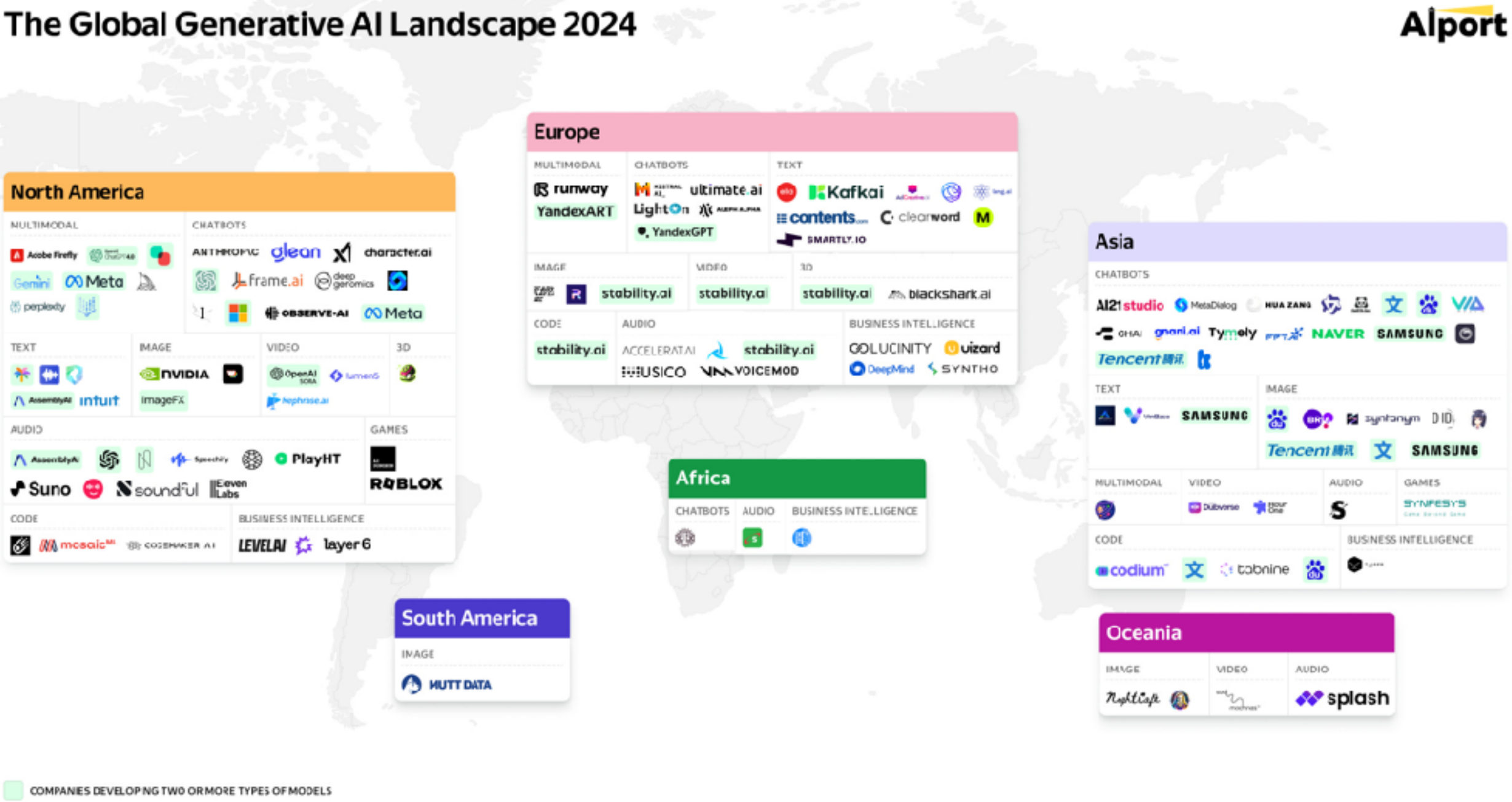
Generative AI Roadmap



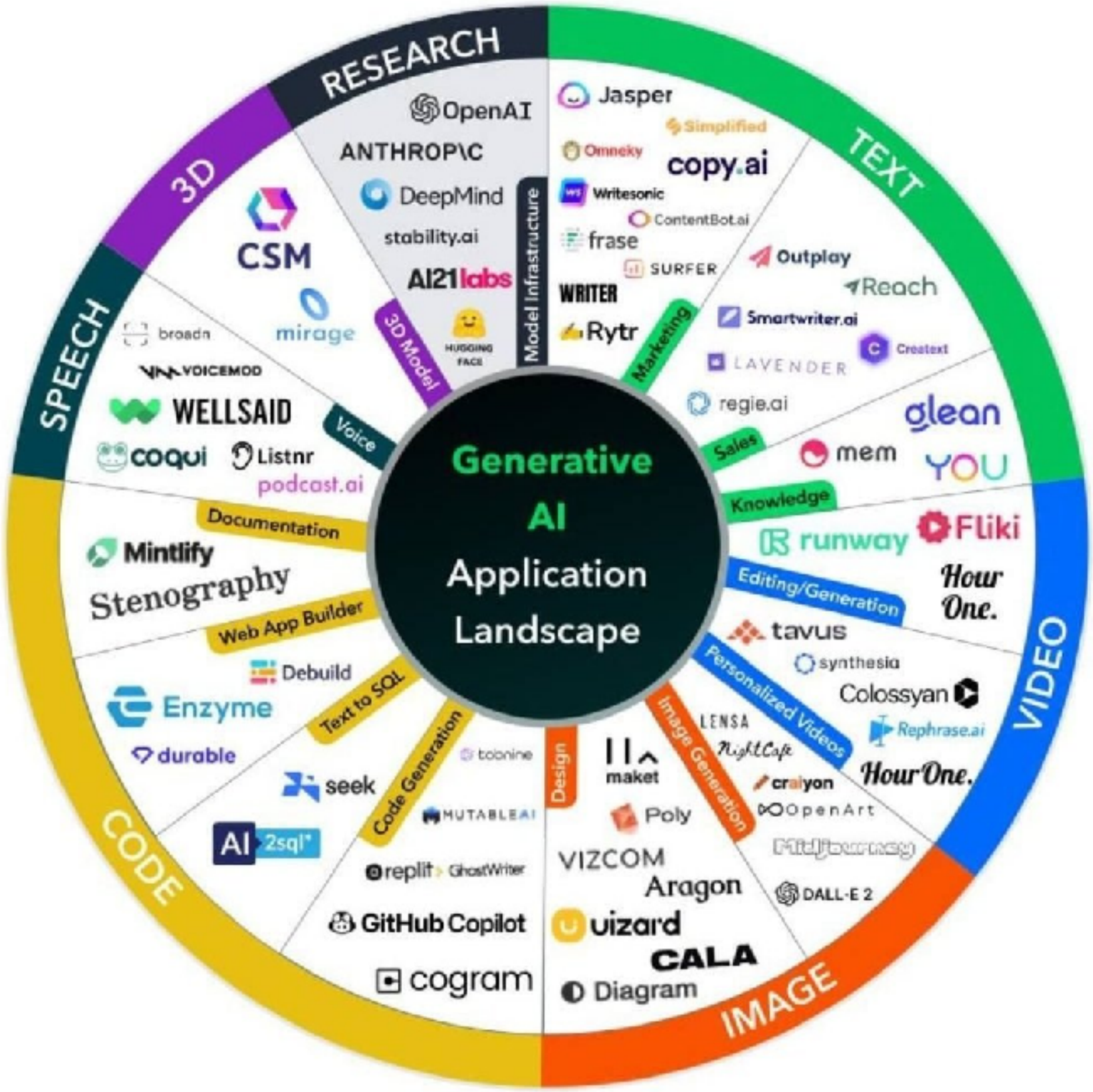
https://roadmap.sh/ai

Application
Tool chains

Global Generative AI Landscape



https://www.blog.aiport.tech/p/the-first-truly-global-generative



Tool chains category

Assist tasks

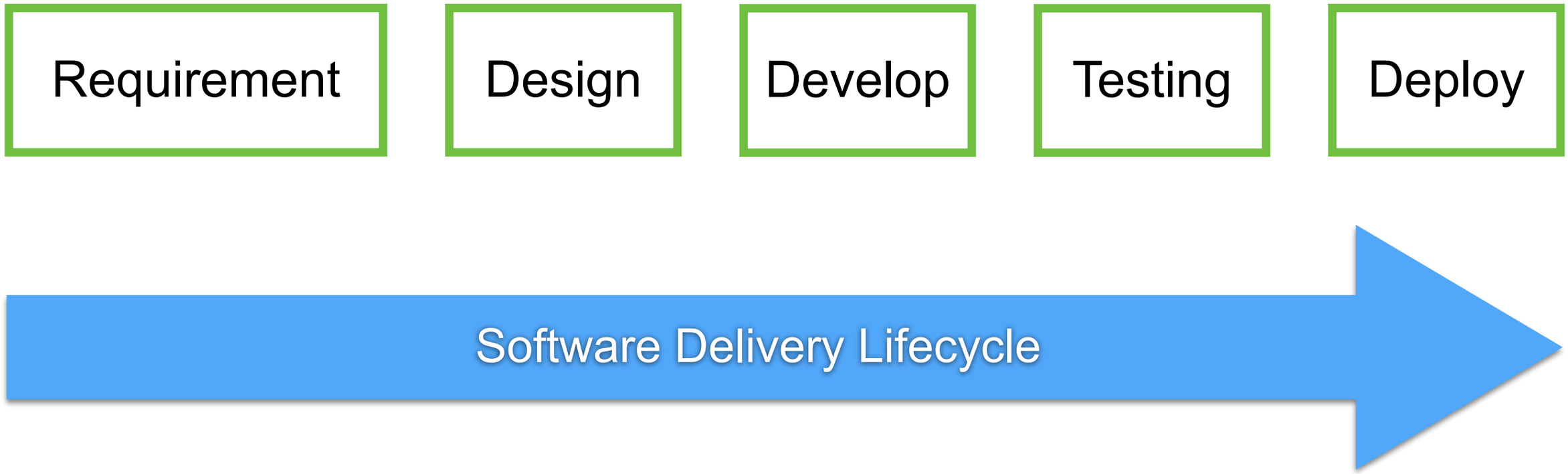
Interaction modes

Prompt composition

Properties of model

Assist tasks

Finding information faster in context
Generating code
Reasoning about code
Transforming code into something ..

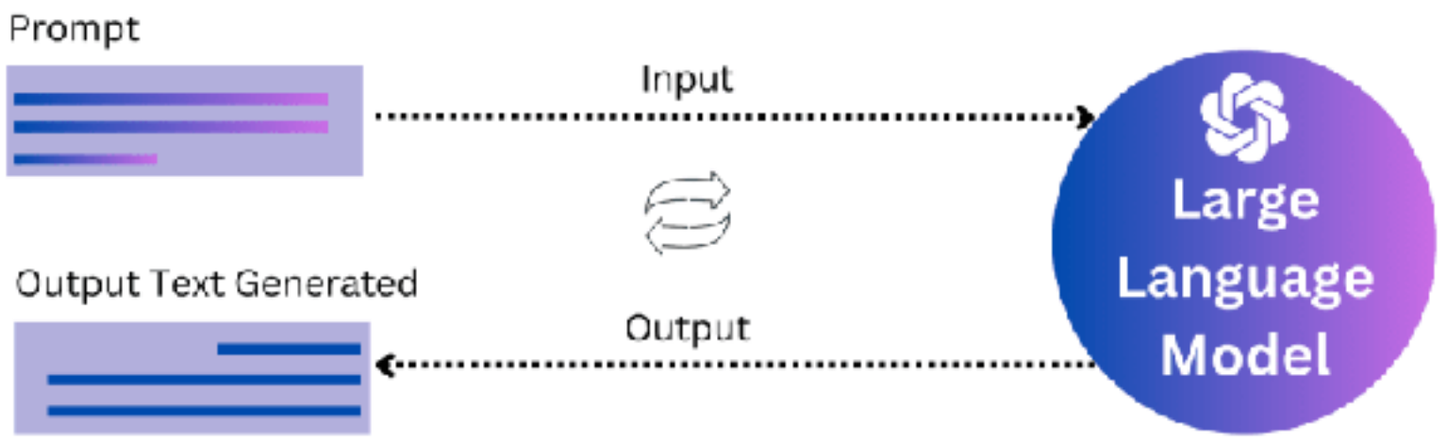


Interaction modes

Chat interfaces
In-line assistance (typing in code editor)
CLI (command-line interface)

Prompt composition

Prompt engineering
Compose prompts from user inputs and context



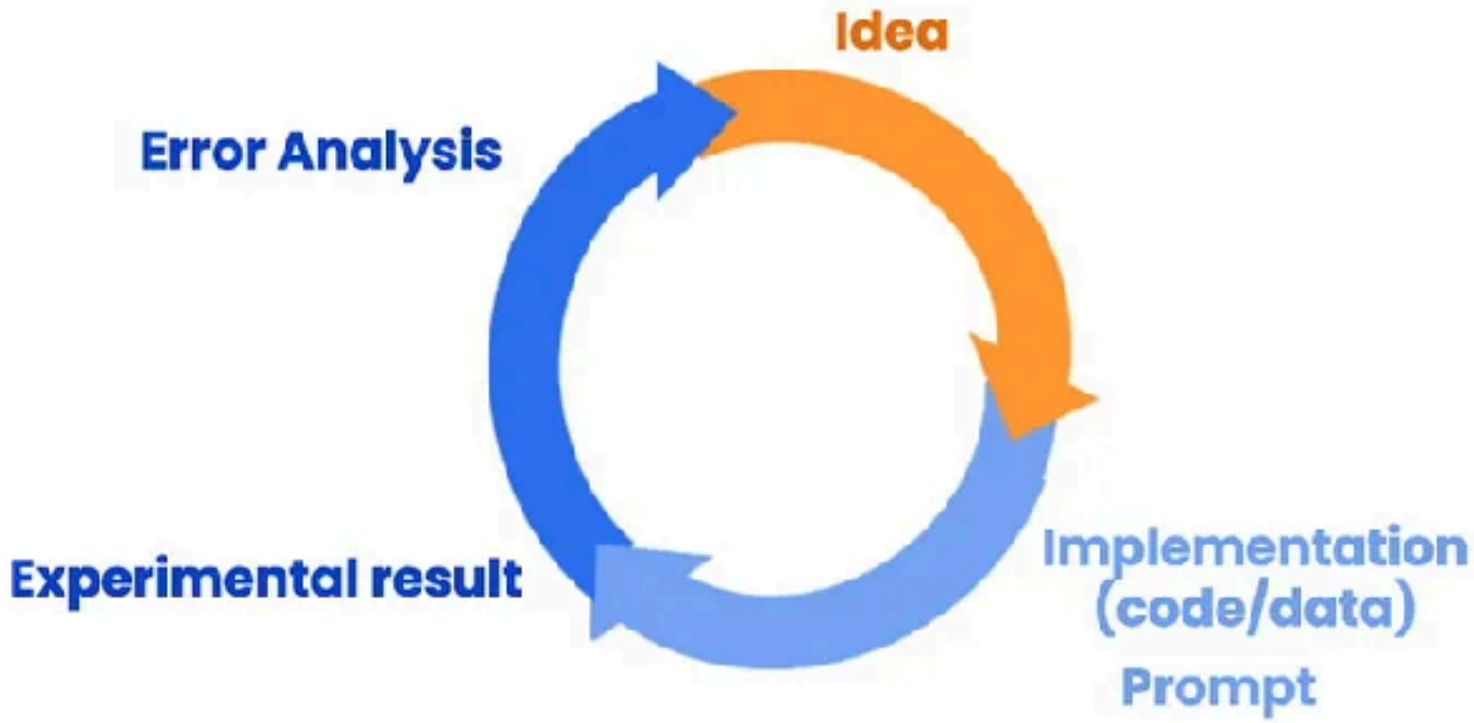
https://platform.openai.com/docs/guides/prompt-engineering

Better Prompt

Write clear instructions
Provide reference text
Split complex tasks into simpler subtasks
Give the model time to think
Testing and improve …

https://platform.openai.com/docs/guides/prompt-engineering

Iterative Prompt Development

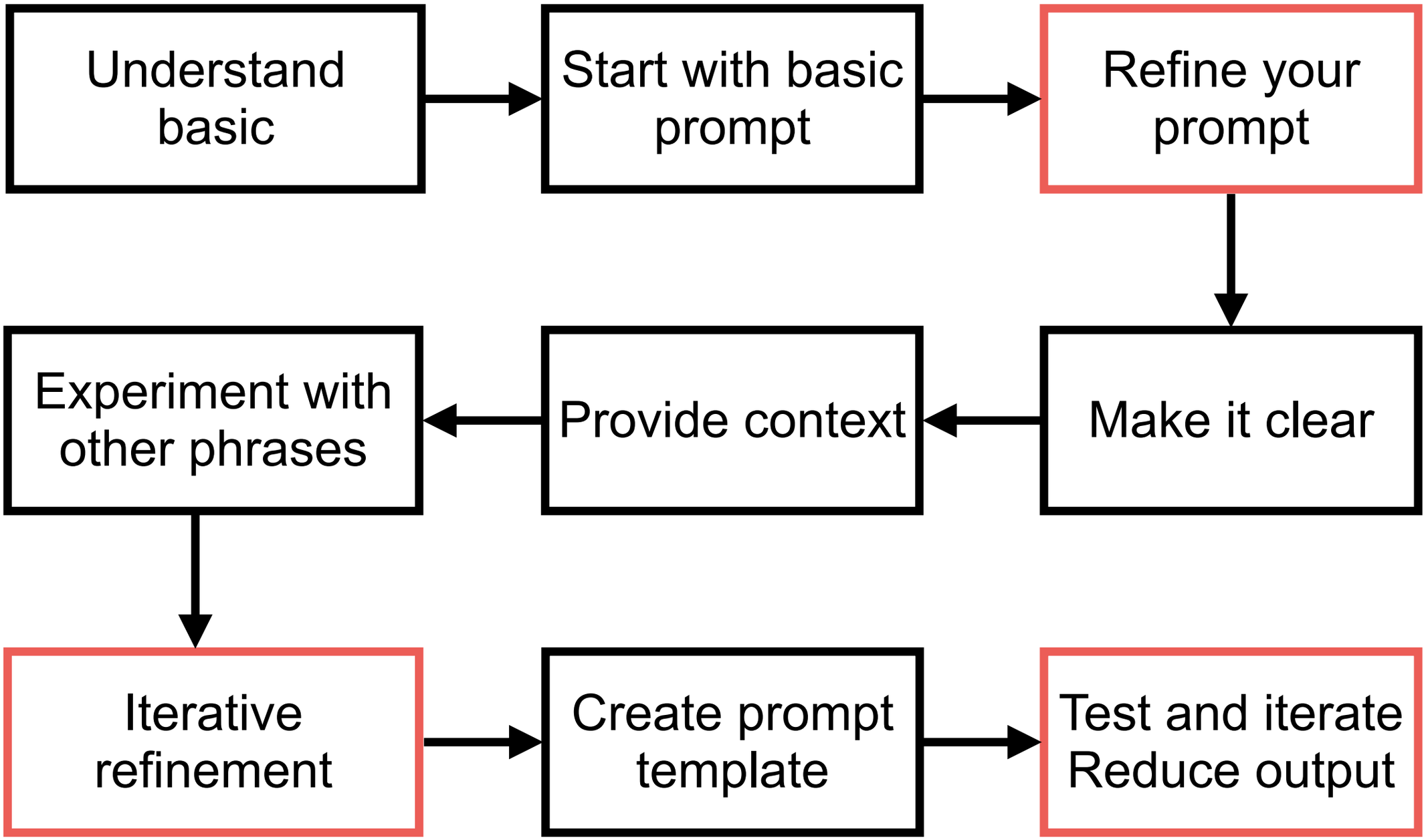


Iterative Process

- Try something
- Analyze where the result does not give what you want
- Clarify instructions, give more time to think
- Refine prompts with a batch of examples

https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/

Basic of Prompt Engineer

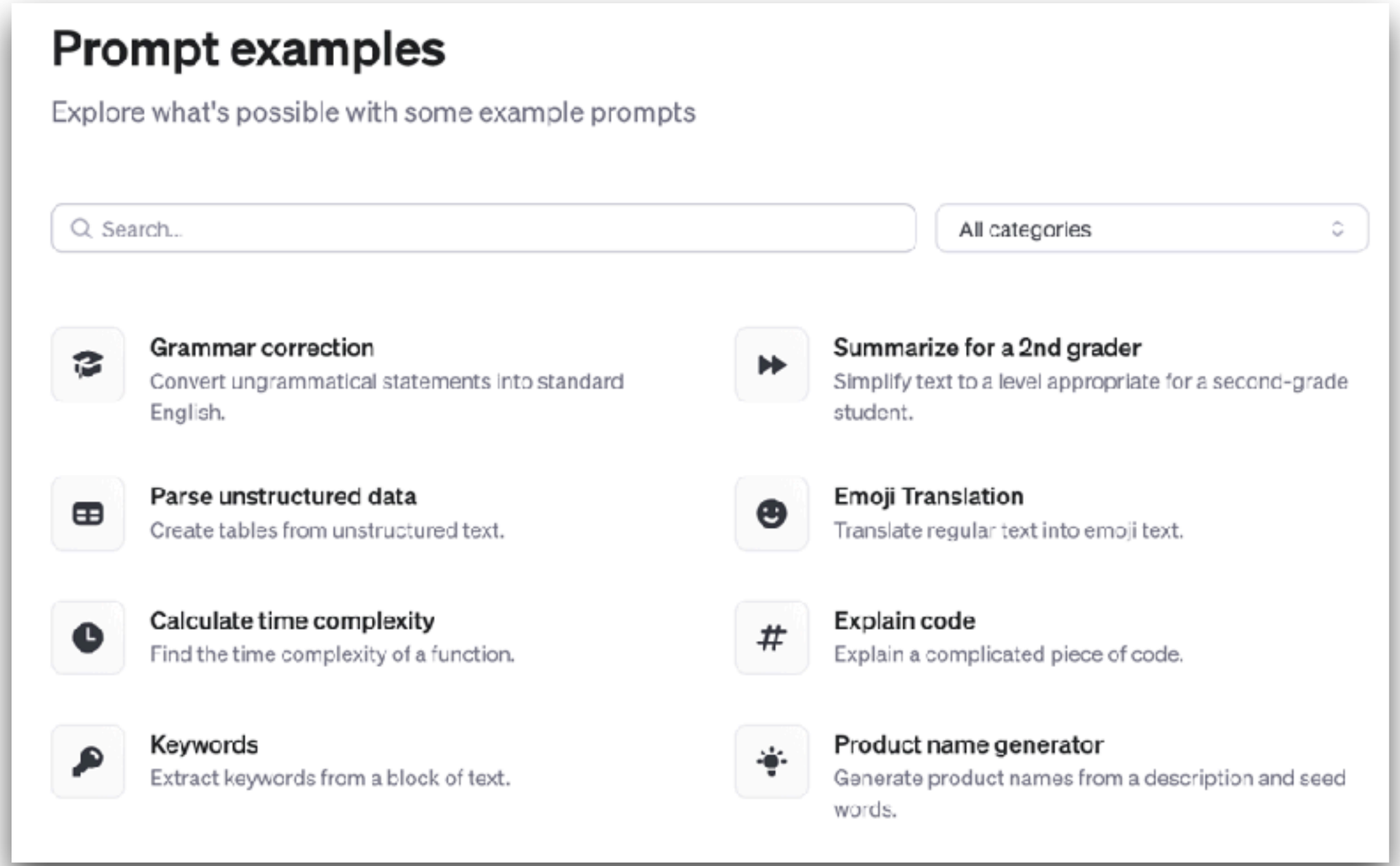


Better Prompt

Write clear instructions
Provide reference text
Split complex tasks into simpler subtasks
Give the model time to think
Testing and improve …

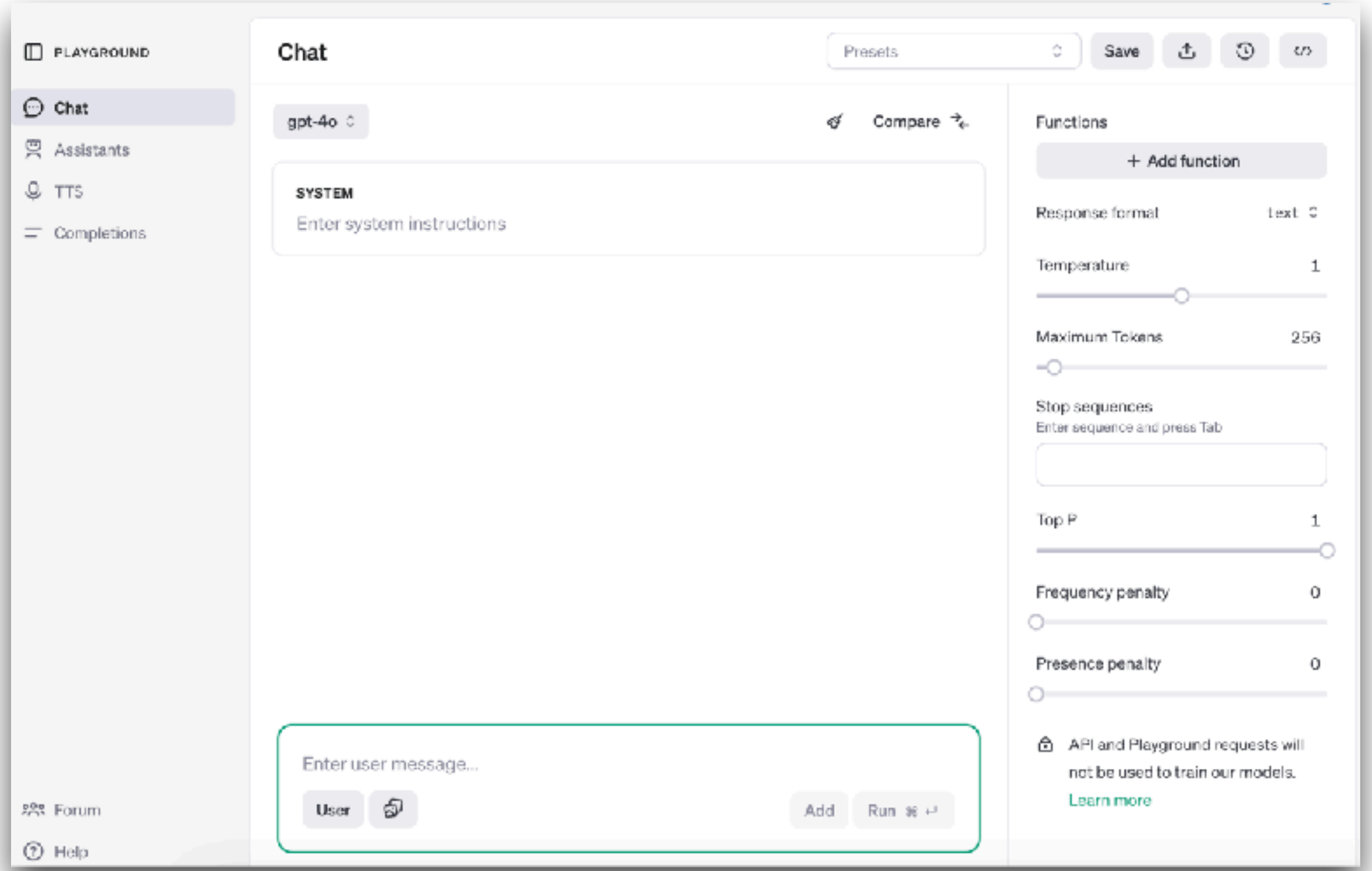
https://platform.openai.com/docs/guides/prompt-engineering/six-strategies-for-getting-better-results

Example Prompt



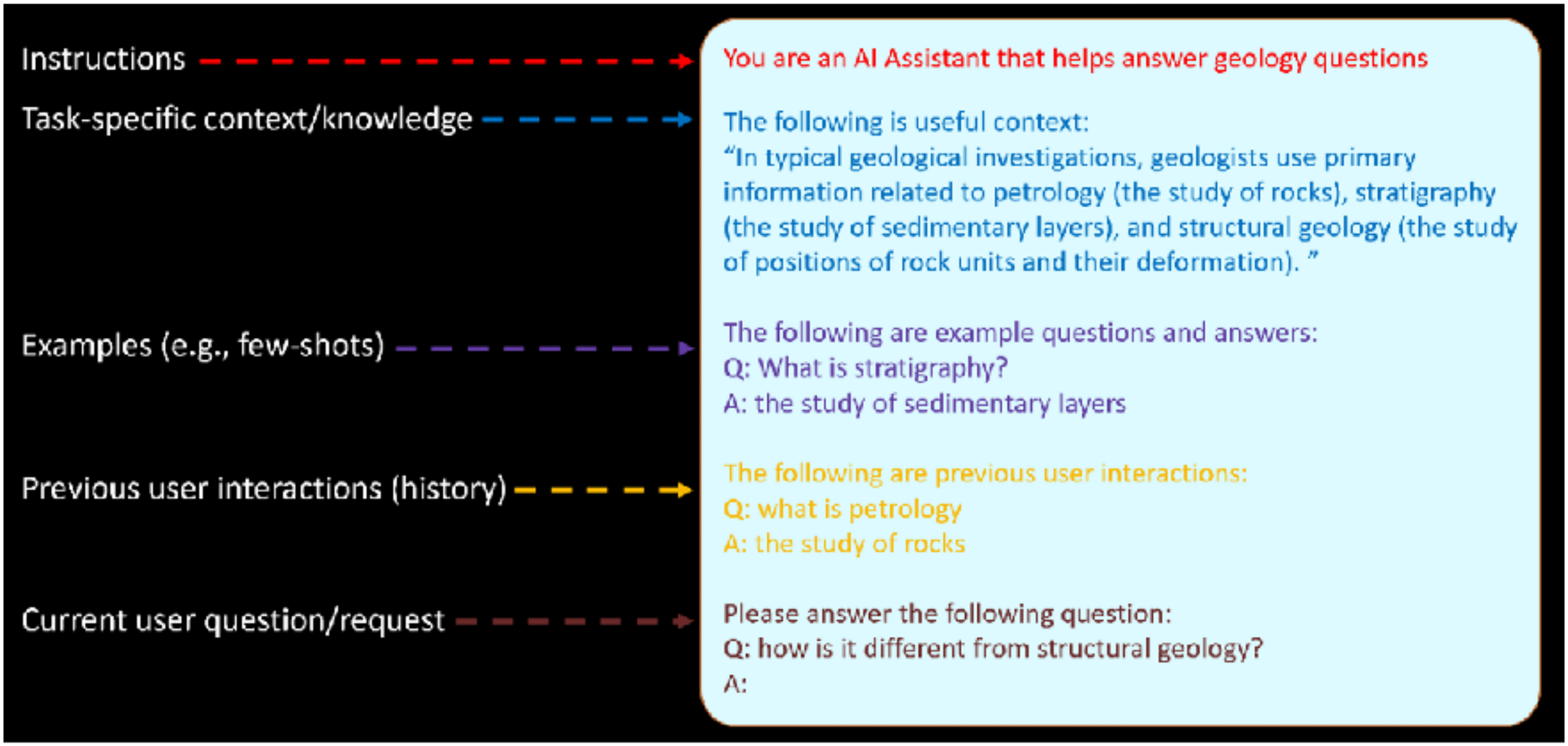
https://platform.openai.com/docs/examples

OpenAI Playground



https://platform.openai.com/playground

Prompt Structure



https://learn.microsoft.com/en-us/ai/playbook/technology-guidance/generative-ai/working-with-llms/prompt-engineering

Prompting Guide

Prompt Engineering

Prompt Engineering Guide

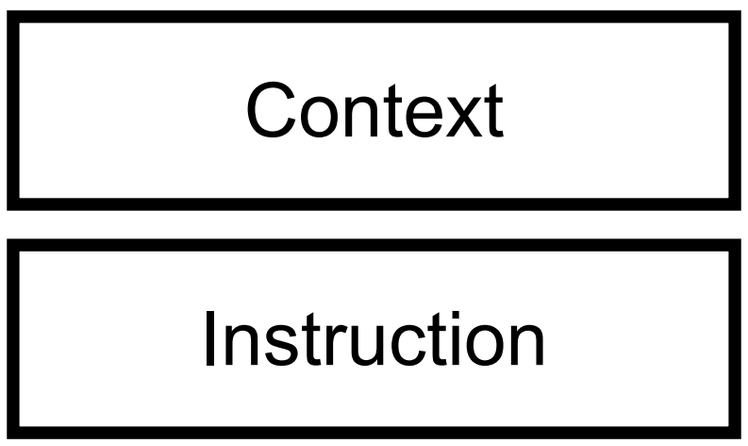
Prompt engineering is a relatively new discipline for developing and optimizing prompts to efficiently use language models (LMs) for a wide variety of applications and research topics. Prompt engineering skills help to better understand the capabilities and limitations of large language models (LLMs).

Researchers use prompt engineering to improve the capacity of LLMs on a wide range of common and complex tasks such as question answering and arithmetic reasoning. Developers use prompt engineering to design robust and effective prompting techniques that interface with LLMs and other tools.

Prompt engineering is not just about designing and developing prompts. It encompasses a wide range of skills and techniques that are useful for interacting and developing with LLMs. It's an important skill to interface, build with, and understand capabilities of LLMs. You can use prompt engineering to improve safety of LLMs and build new capabilities like augmenting LLMs with domain knowledge and external tools.

https://www.promptingguide.ai/

Structure of Prompt



https://platform.openai.com/docs/guides/prompt-engineering

Structure of Prompt !!

APE
Action, Purpose, Expectation

RACE
Role, Action, Context, Expectation

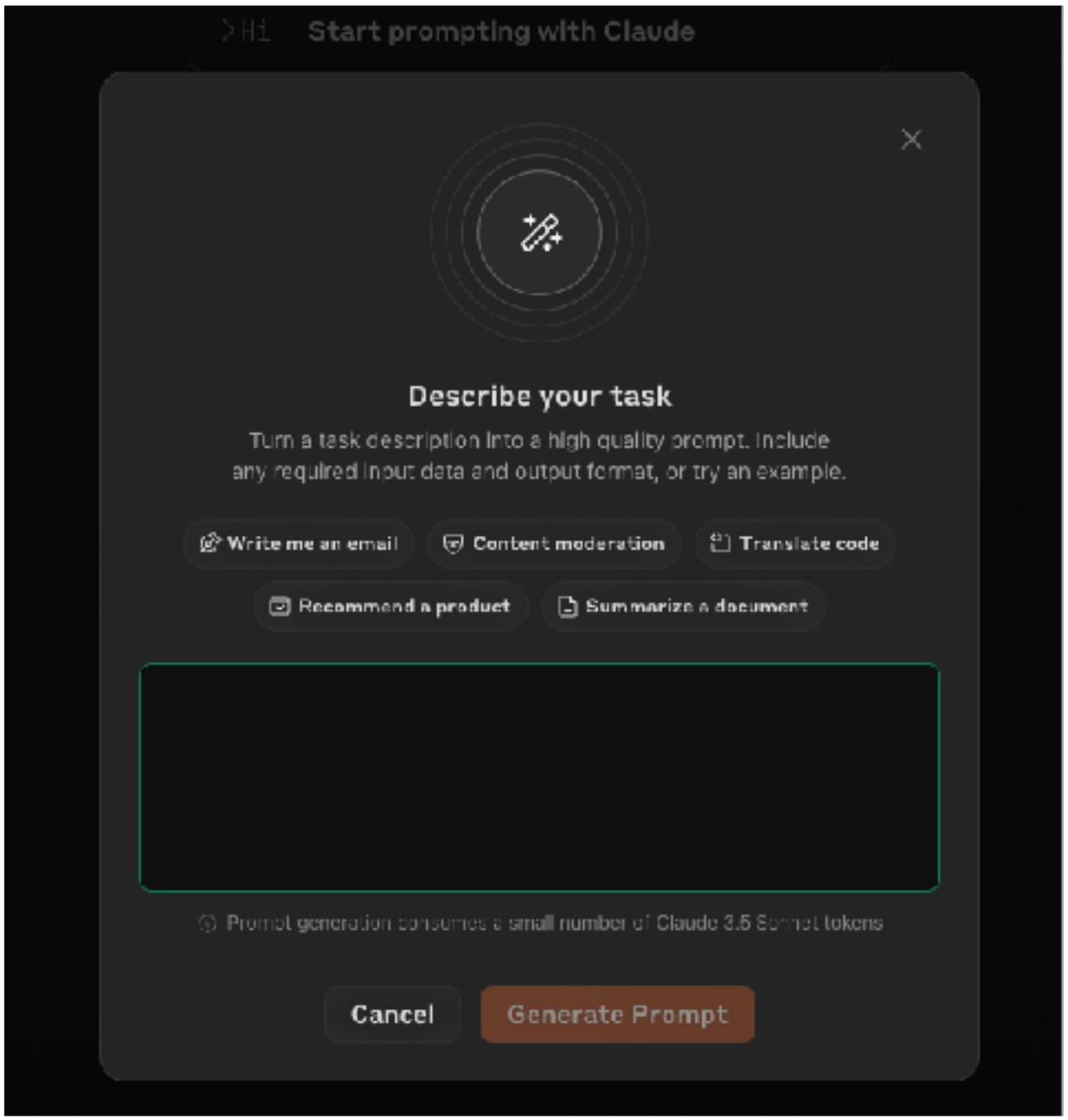
TAG
Task, Action, Goal

COAST
Context, Objective, Action, Scenario, Task

RISE
Role, Input, Step, Expectation

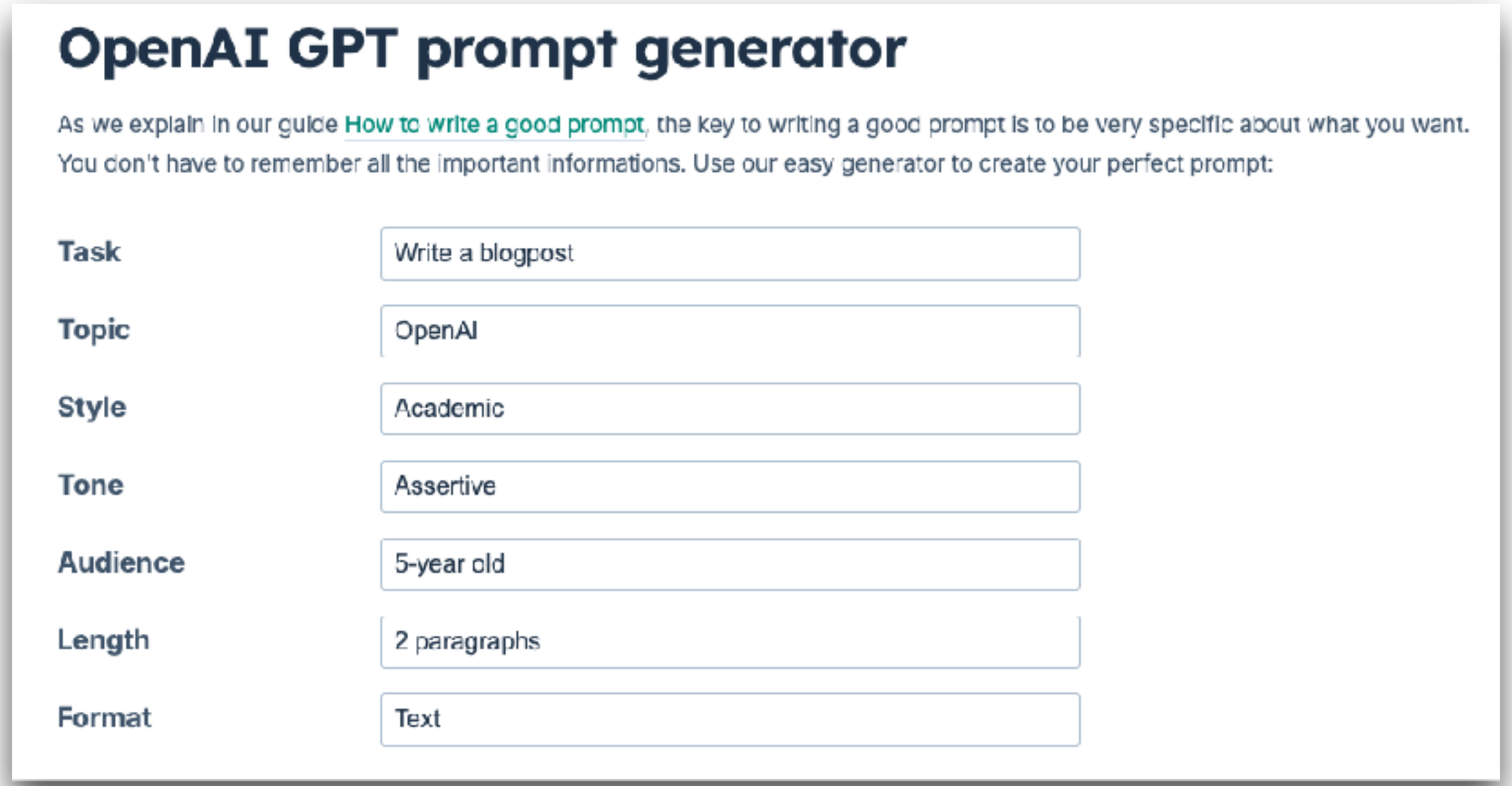
https://twitter.com/pradeepeth/status/1673271866696544257

Prompt Generator



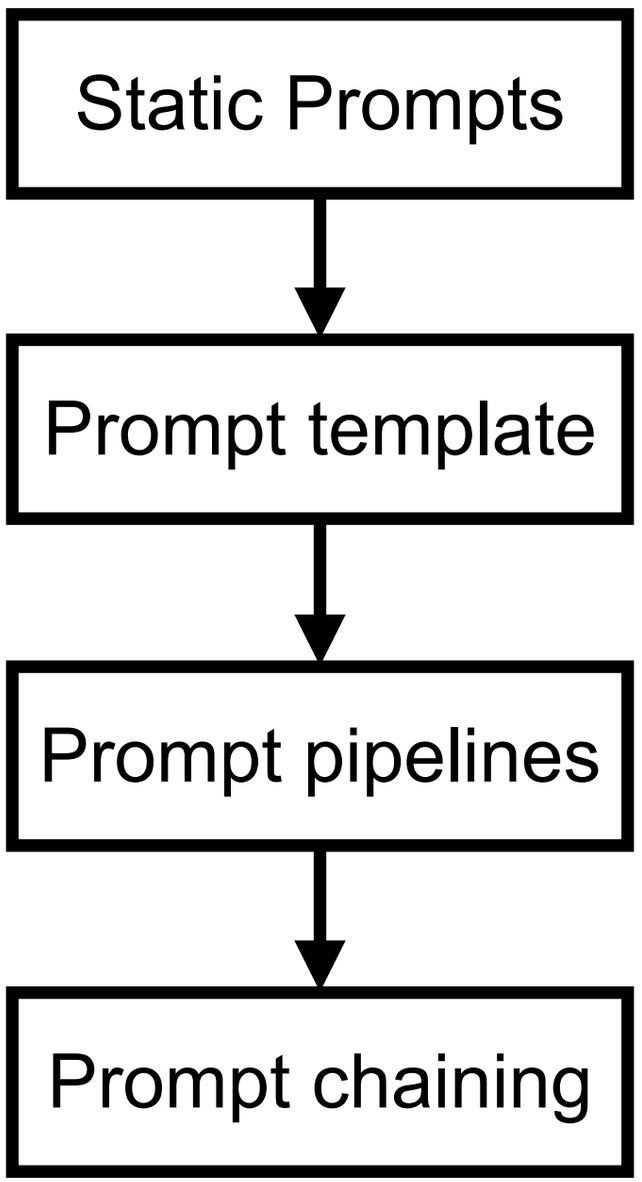
https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/prompt-generator

Prompt Generator



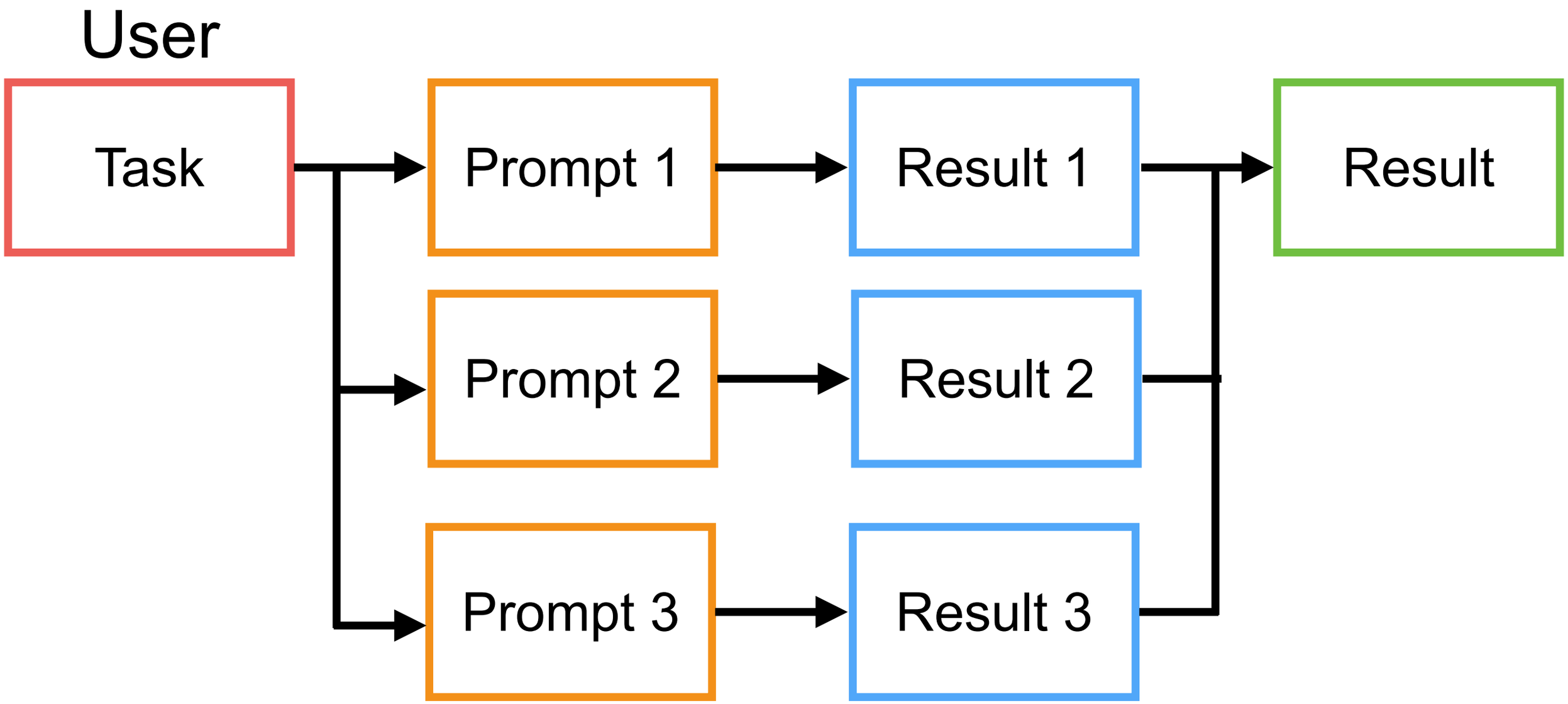
https://gptforwork.com/tools/prompt-generator

Evolution of Prompt Engineering

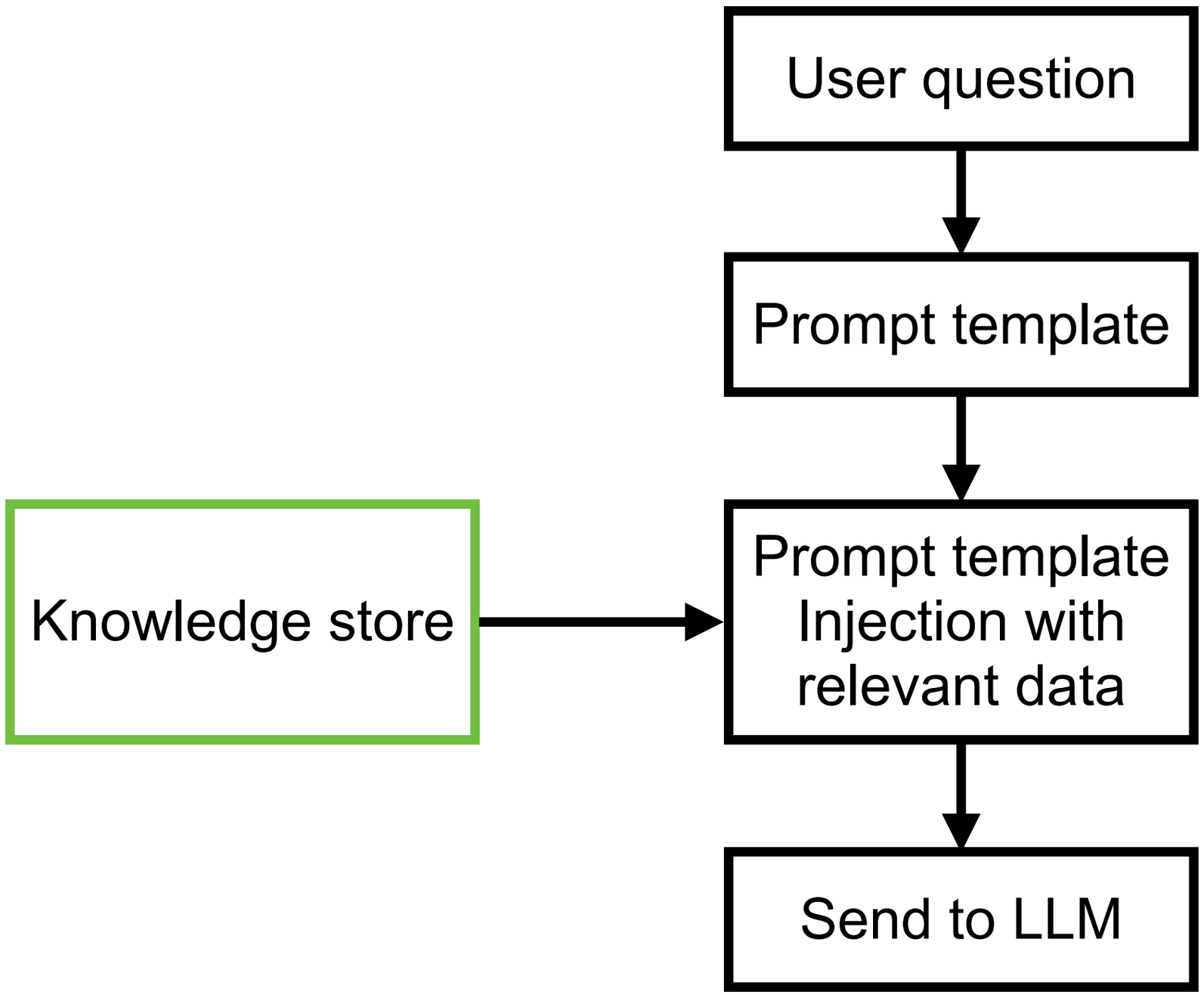


https://github.com/promptslab/Awesome-Prompt-Engineering

Basic Prompts

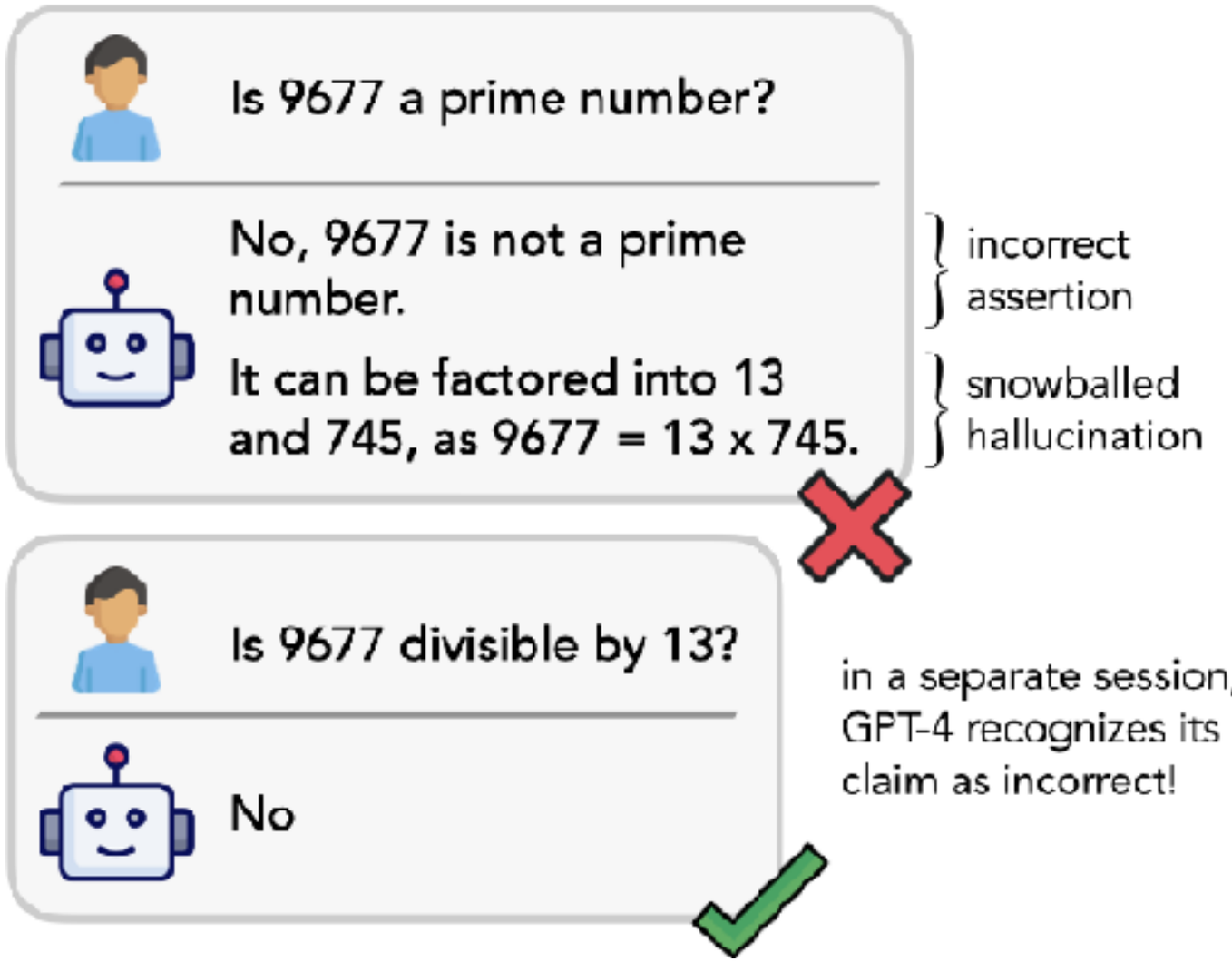


Prompt Pipelines



https://cobusgreyling.medium.com/the-evolution-of-prompt-engineering-29c3d6943af2

Hallucinations in LLMs



Hallucinations in LLMs

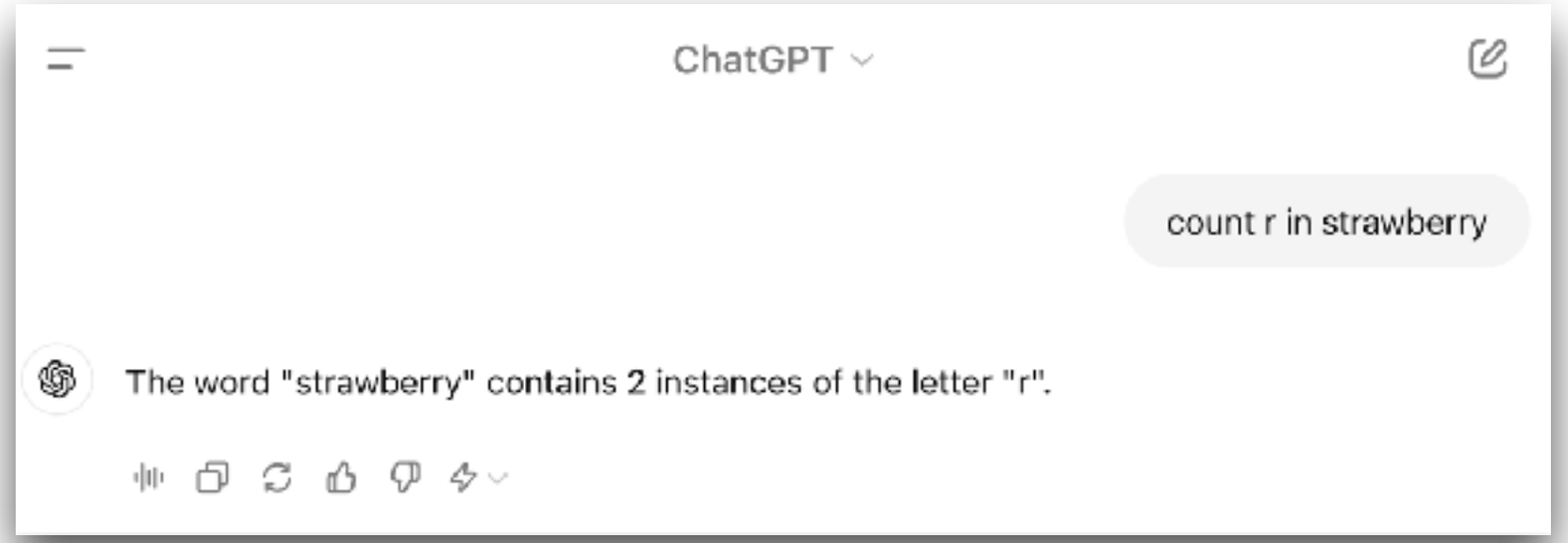
Training data set issue

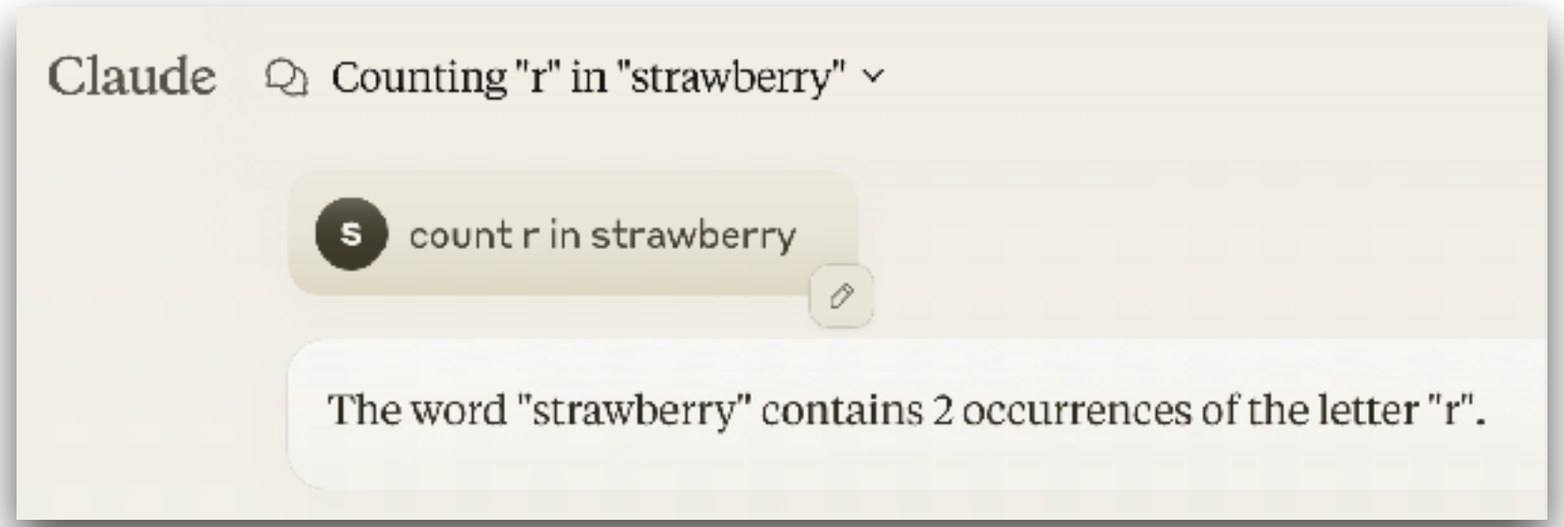
Architecture and training objective

Prompt engineering

Over-optimization

Count r in Strawberry !!





https://hackernoon.com/why-cant-ai-count-the-number-of-rs-in-the-word-strawberry

Count r in Strawberry !!

Perplexity

Counting the Letter "R" in "Strawberry"

To count the number of times the letter "R" appears in the word "strawberry", we can break it down character by character:

S-T-**R**-A-W-B-E-**R**-**R**-Y

In this analysis, we see that the letter "R" appears 3 times in the word "strawberry" [1] [2] [3].

Some key points:

- Language models like ChatGPT do not always count letters correctly in words like "strawberry" [2] [3]
- This is because they break down words into tokens rather than individual letters, and the tokens may not map directly to the letters [2]
- To get the right answer, you can have the AI spell out the word and count the "R"s as it goes [4]
- Counting letters is a simple task for humans but can trip up even advanced AI systems [3] [4]

So in summary, the letter "R" appears 3 times in the word "strawberry". While this is obvious to humans, it can be a tricky problem for current AI language models to solve consistently.

Count r in Strawberry !!

Count r in Strawberry

step-by-step of your thinking

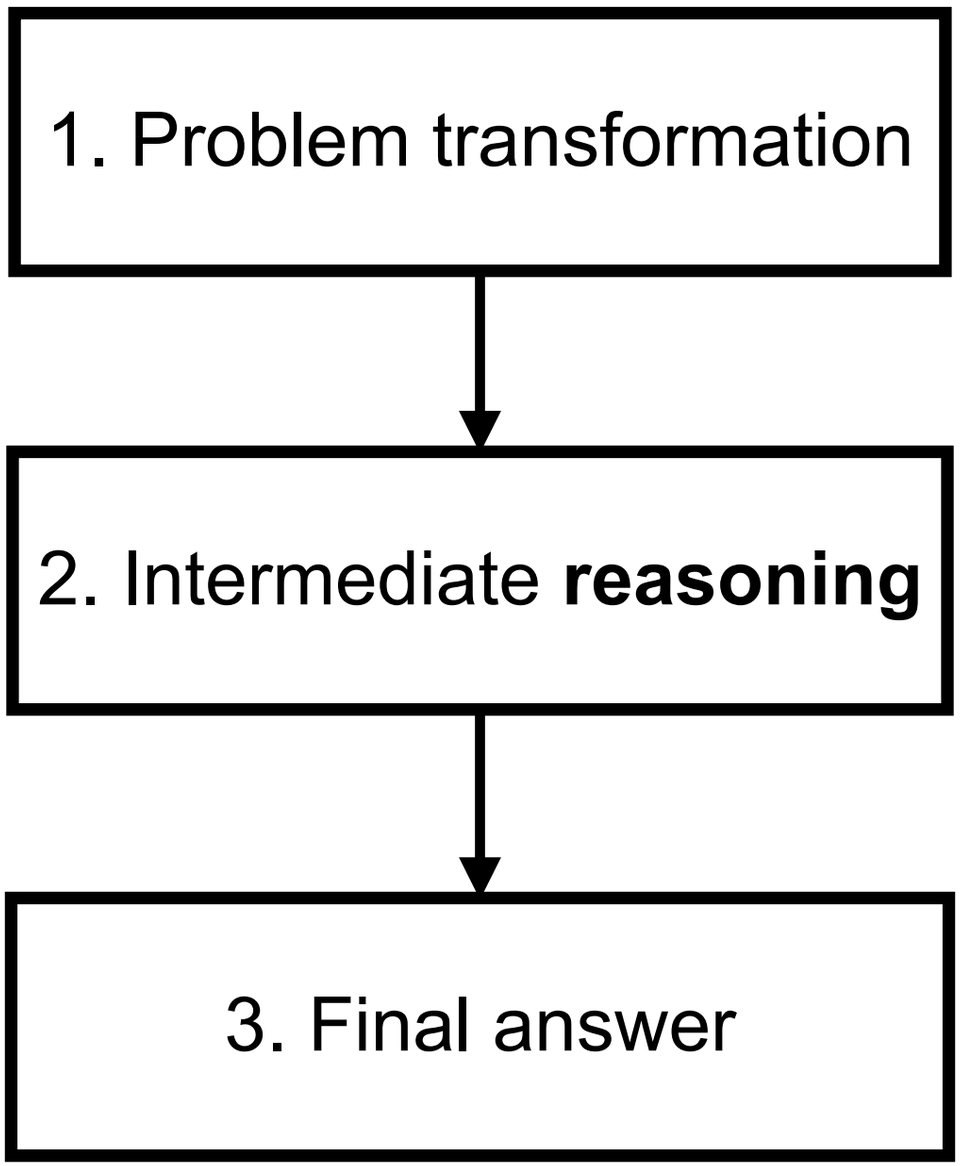
Chain of Thought Prompting (CoT)

Technique used to improve the reasoning ability of LLM

Try to break down a complex problem into smaller, More manageable steps, lead to final answer

OpenAI o1 model

Chain of Thought Prompting (CoT)



Chain of Thought Prompting (CoT)

(a) Few-shot

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The answer is 8. ✗

(b) Few-shot-CoT

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?
A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. 5 + 6 = 11. The answer is 11.

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A:

(Output) The juggler can juggle 16 balls. Half of the balls are golf balls. So there are 16 / 2 = 8 golf balls. Half of the golf balls are blue. So there are 8 / 2 = 4 blue golf balls. The answer is 4. ✓

(c) Zero-shot

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: The answer (arabic numerals) is

(Output) 8 ✗

(d) Zero-shot-CoT (Ours)

Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: ***Let's think step by step.***

(Output) There are 16 balls in total. Half of the balls are golf balls. That means that there are 8 golf balls. Half of the golf balls are blue. That means that there are 4 blue golf balls. ✓

https://www.promptingguide.ai/techniques/cot

Advice from OpenAI (o1 model)

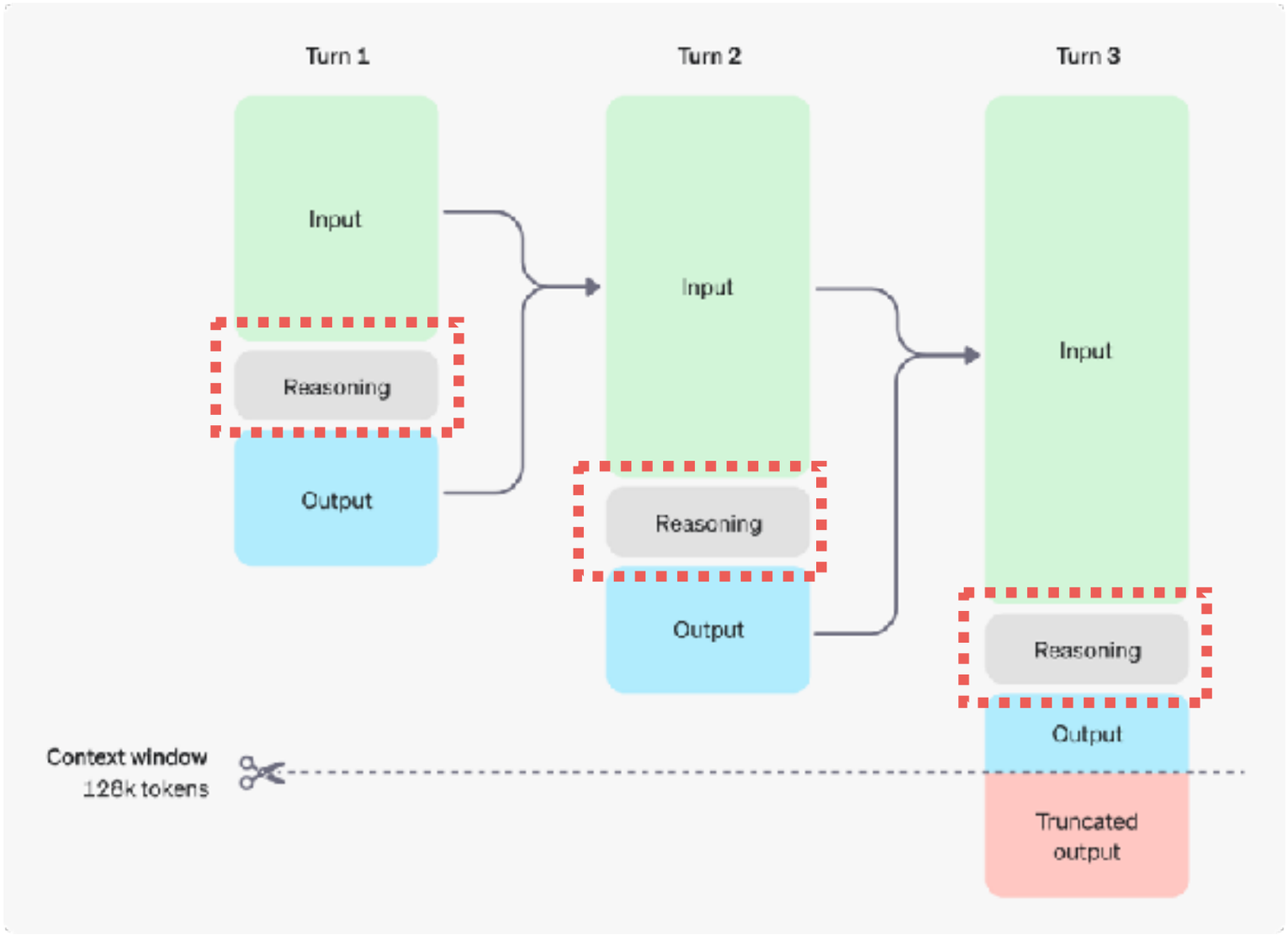
CoT prompt may not enhance performance

Keep prompts simple and direct	Avoid CoT
Use delimiter for clarity	Limit additional context in RAG

https://platform.openai.com/docs/guides/reasoning/advice-on-prompting

How reason works ?

Reasoning token (invisible token but billed)



https://platform.openai.com/docs/guides/reasoning/advice-on-prompting

Write code to generate prompt

Retrieval Augmented Generation (RAG)

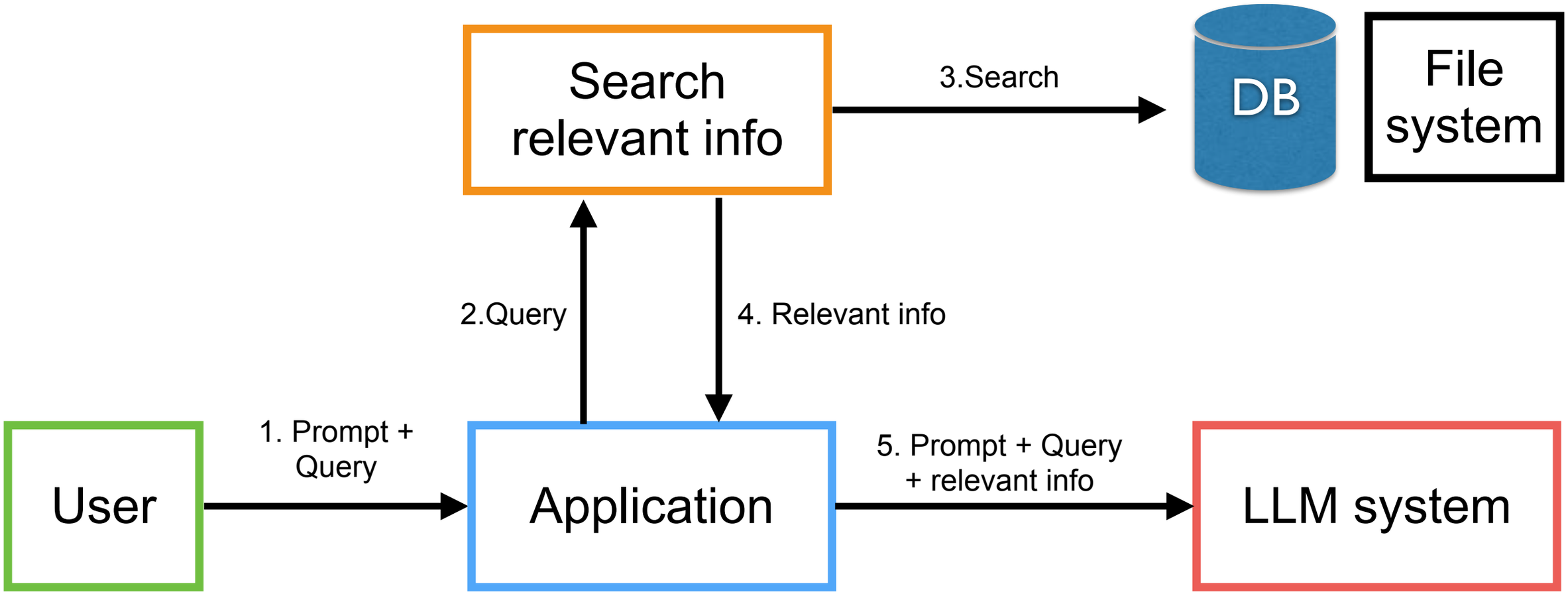
What is RAG ?

Enhance LLM with external knowledge
Improve your LLM models, more accurate answer

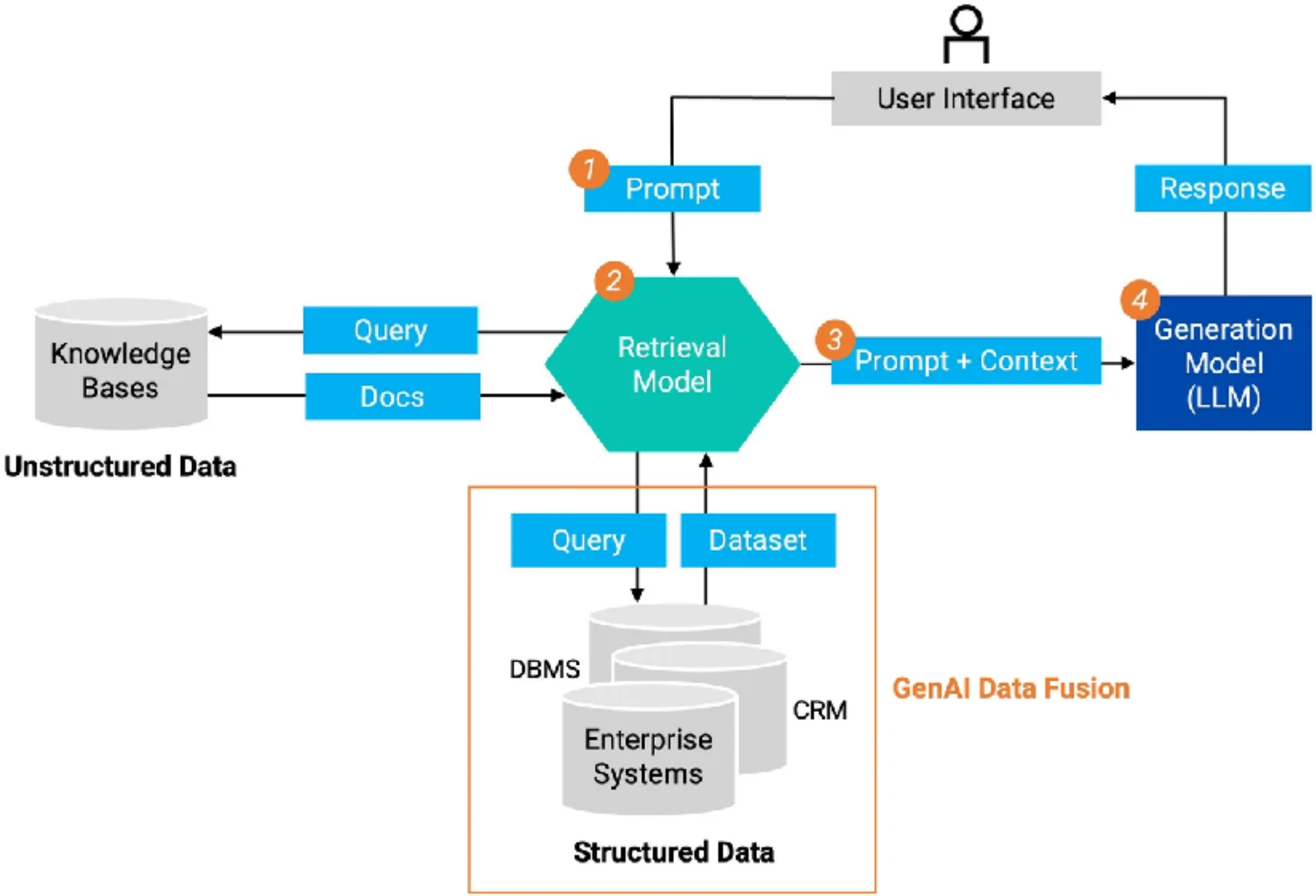
Proprietary knowledge	Up-to-date Information
Citing sources	Data security Access control List (ACL)

RAG with LLM

Improve your LLM models, more accurate answer

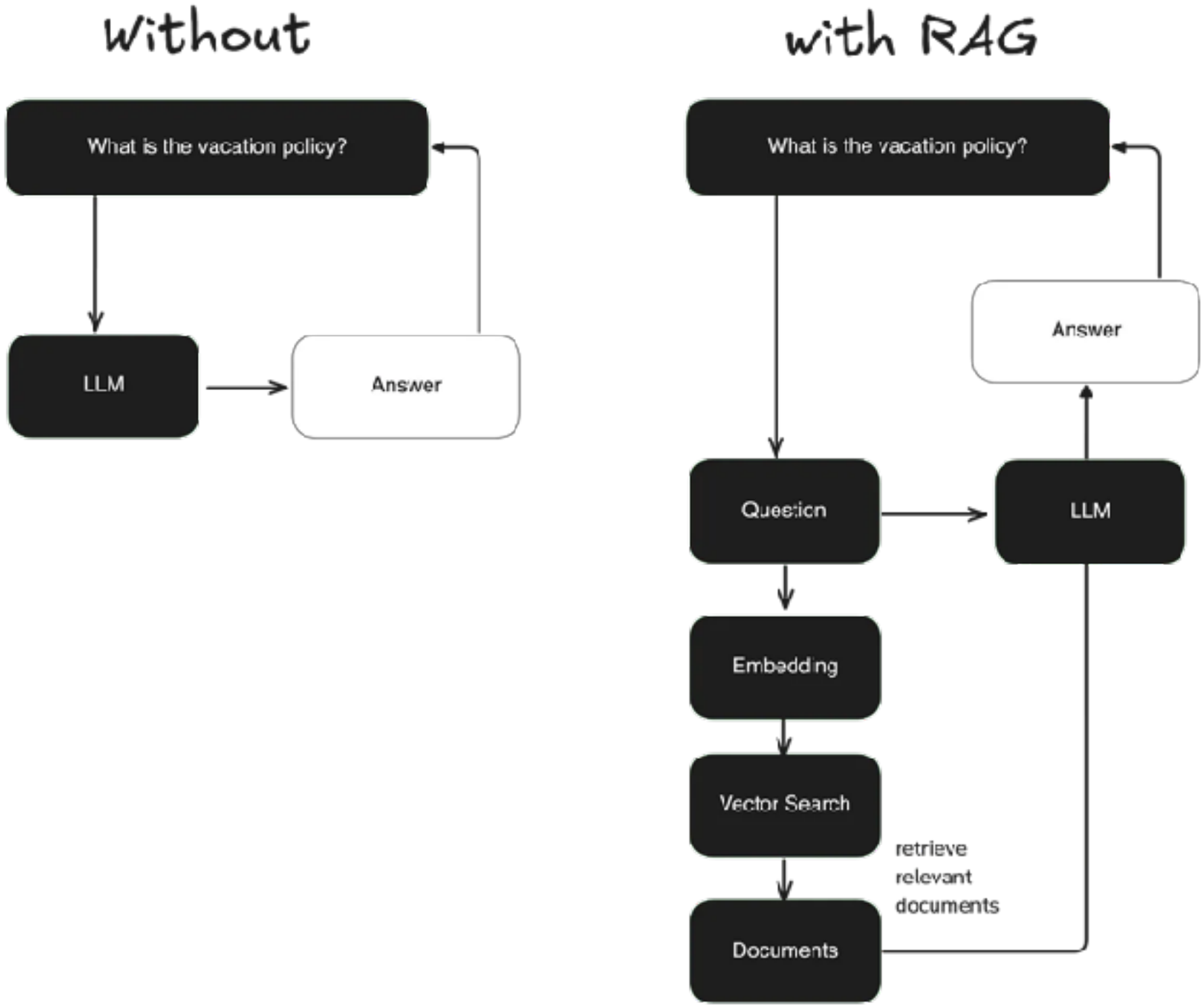


Retrieval-Augmented Generation (RAG) Framework



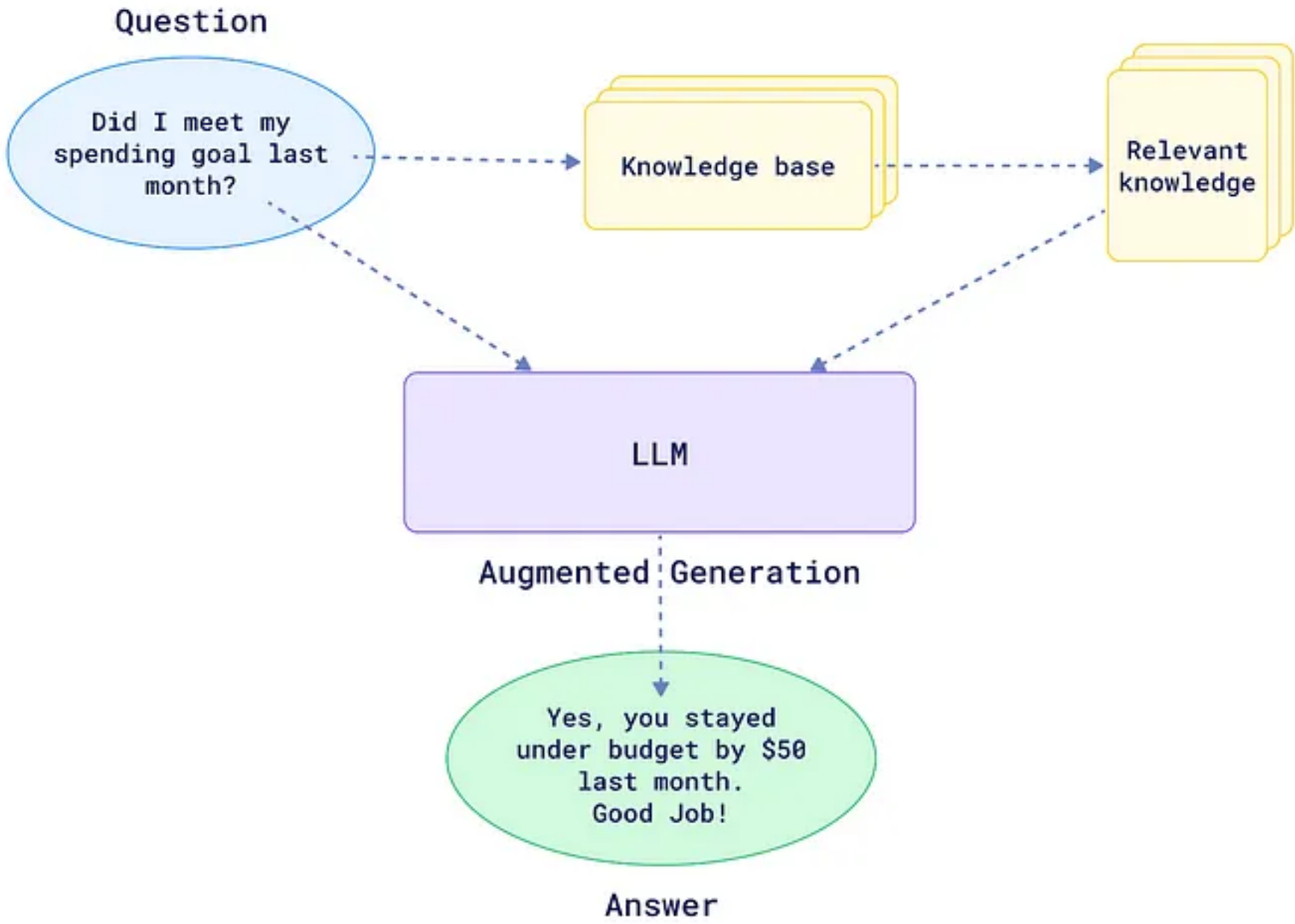
https://www.k2view.com/blog/rag-genai

RAG ?

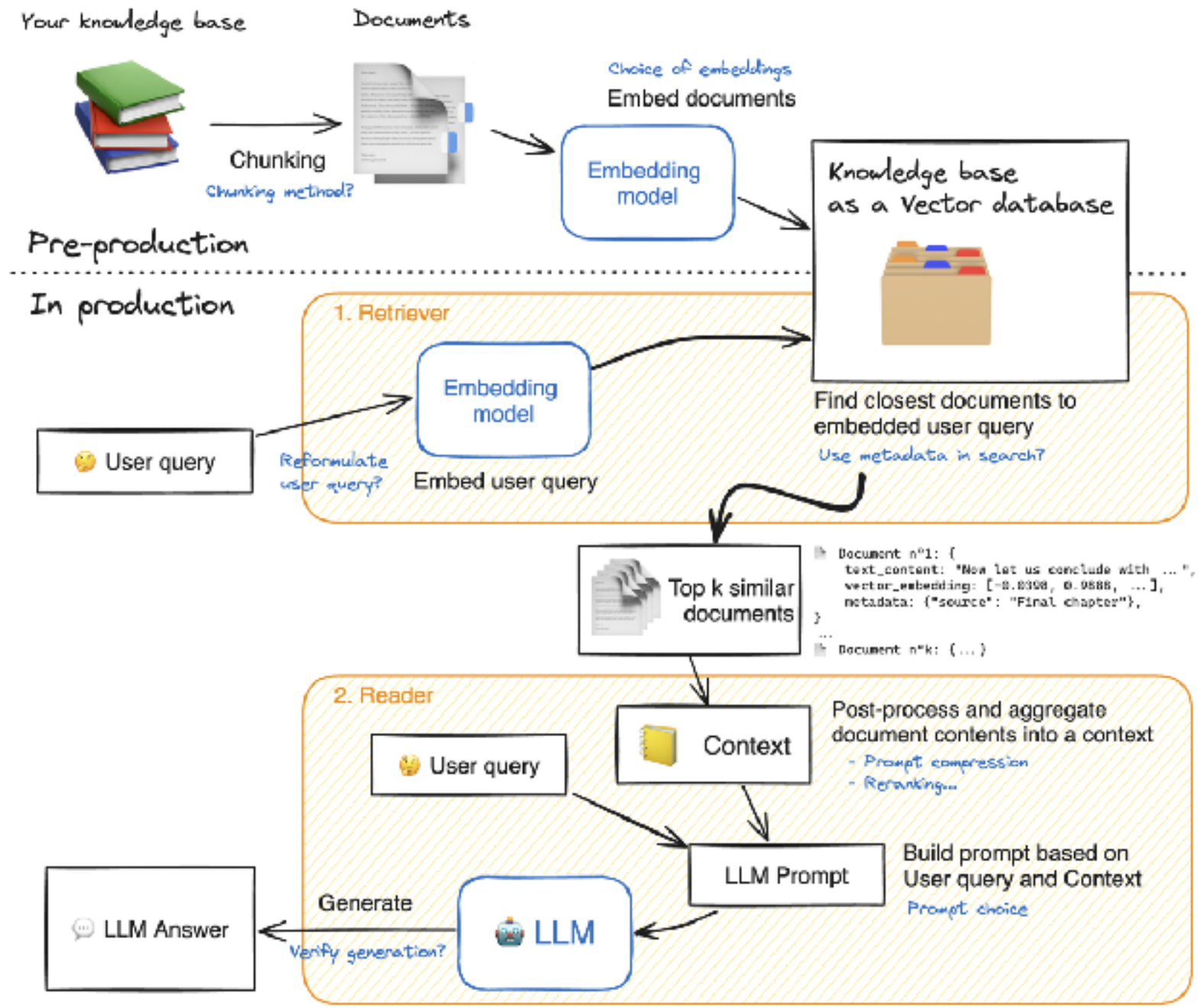


https://medium.com/google-cloud/google-cloud-rag-api-c7e3c9931b3e

RAG ?

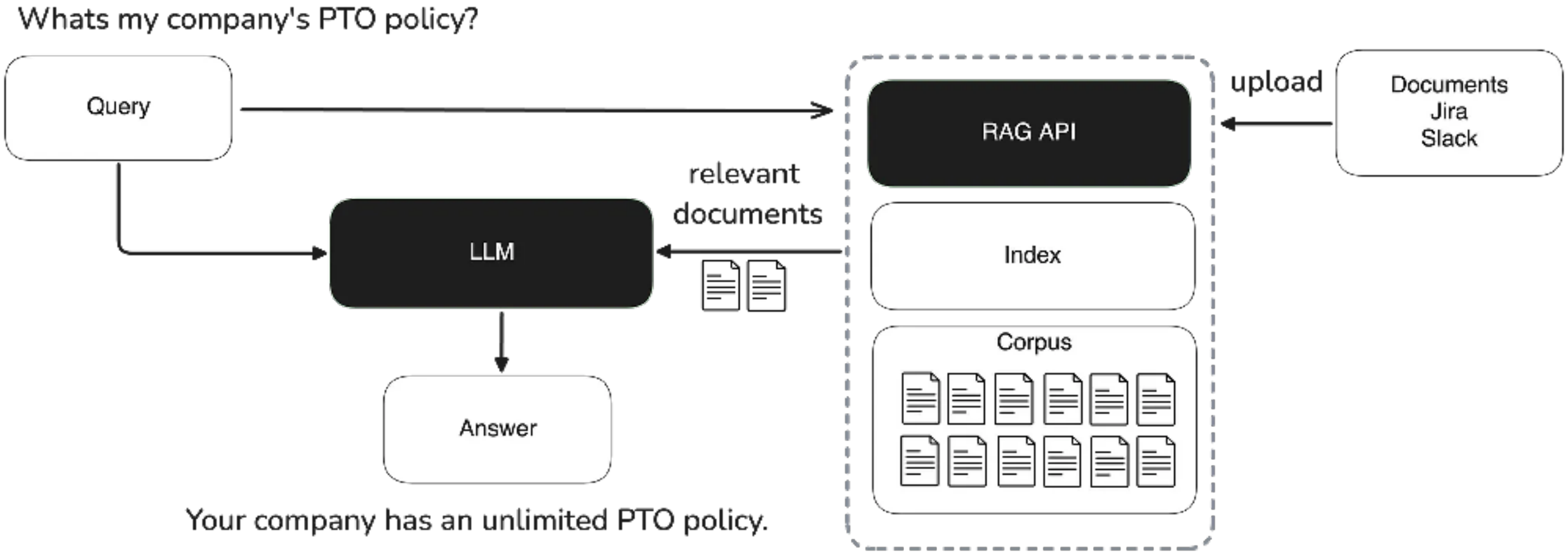


RAG ?



https://huggingface.co/learn/cookbook/advanced_rag

Google RAG APIs



https://medium.com/google-cloud/google-cloud-rag-api-c7e3c9931b3e

RAG Key Terms

Tokenization

Chunking

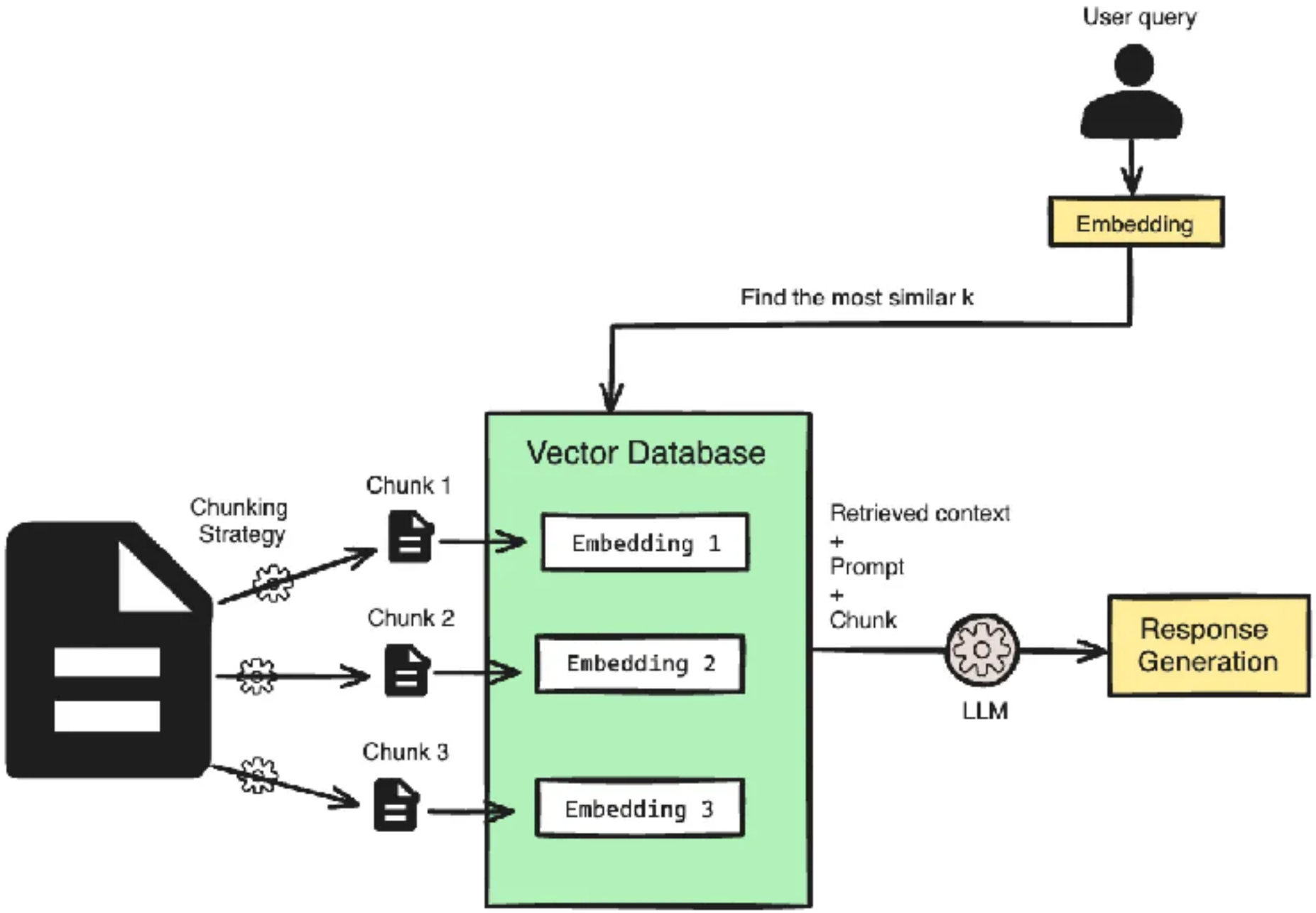
Embedding

Embedding models

Similarity search
Vector search

LLM window context

RAG Key Terms



https://levelup.gitconnected.com/semantic-chunking-for-enhanced-rag-applications-b6bc92942af0

Tokenization

Process of breaking down text into smaller units called **tokens**

Tokens can be words, characters, or subwords, **depending on the task**

Help prepare text data for analysis by representing it in a structured way

Chunking

Chunking is the **grouping of tokens into meaningful sections**, often based on grammatical structure, like noun phrases or verb phrases.

Helps in **understanding** the structure of sentences and the relationships between words

Chunking Strategies !!

Fixed length

Recursive characters

Document-based

Semantic

Phrase-based on linguistic patterns

https://www.linkedin.com/pulse/prompt-engineering-chunking-strategies-ravi-naarla/

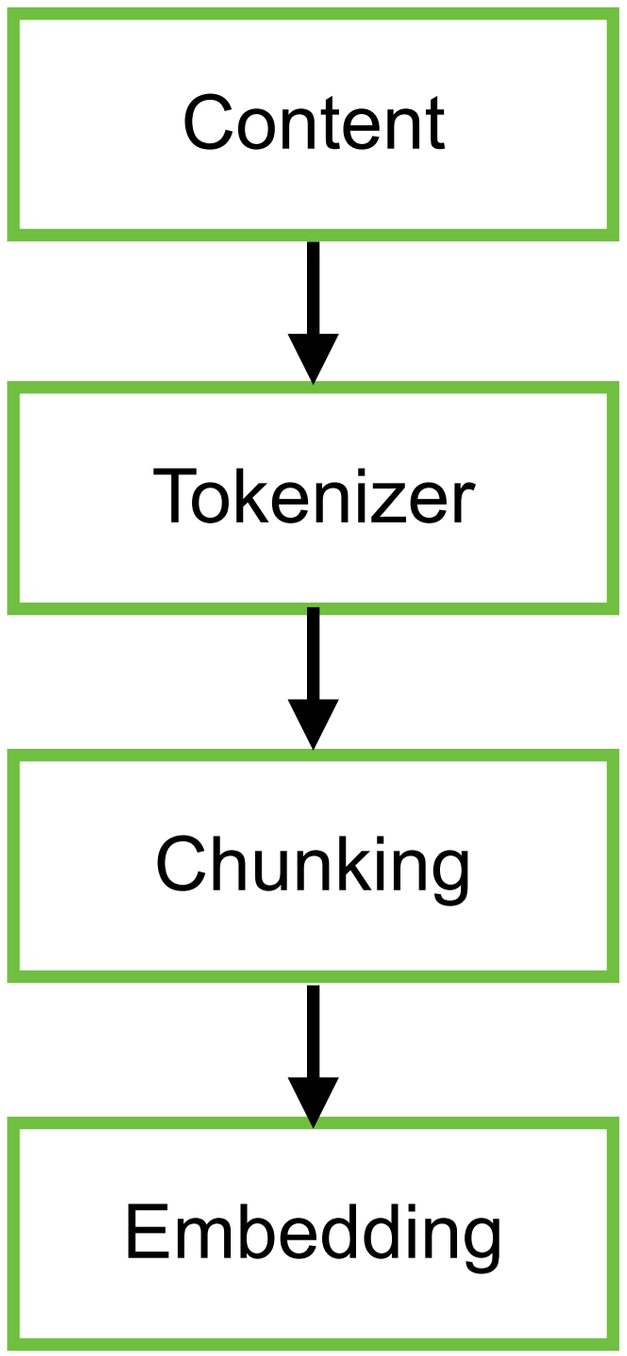
Embedding

Embedding is a representation of text data in **numerical vector form**

Each word or sentence is represented as a **high-dimensional vector**, capturing its semantic meaning

Embeddings allow text to be used in **machine learning models** by converting it into a format the models can process.

Summary



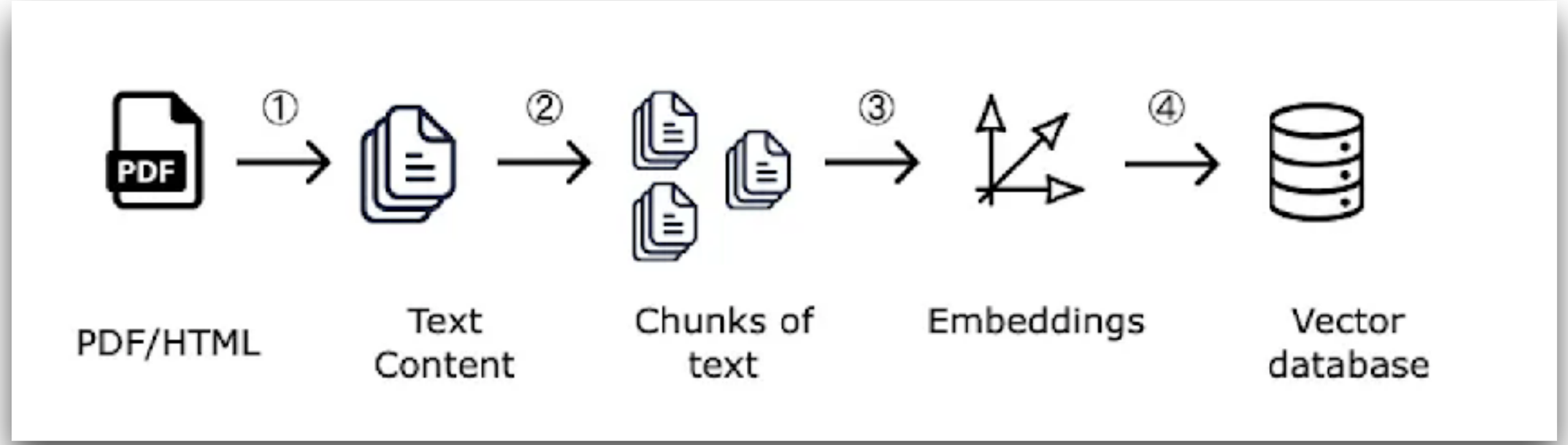
Hello world

Token	Hello	World
Token id	1	2

[1,2]

```
[[-3.95261124e-02 -1.66764930e-02  9.81786475e-02  1.27390521e-02
  -4.50244620e-02 -1.57272965e-02  6.66662678e-02  2.10209060e-02
   9.99843255e-02  3.95087115e-02  3.87372263e-02 -2.51084827e-02
   5.32396603e-03  4.54628207e-02  7.42979953e-03  2.00463505e-03
  -4.58151475e-02 -1.20408414e-03 -8.42441767e-02 -3.62469666e-02
  -1.61933392e-01  6.05303086e-02  4.38679755e-03 -1.14583876e-03
  -1.38427421e-01  2.52840482e-02  5.86531451e-03 -8.12693834e-02
   4.02542809e-03 -1.90829430e-02 -1.93073396e-02  9.50673595e-03
  -1.89097337e-02  3.16560529e-02 -4.22104448e-02 -1.75430663e-02
   3.54745984e-02  1.34416856e-02  4.95691150e-02  5.71430475e-02
```

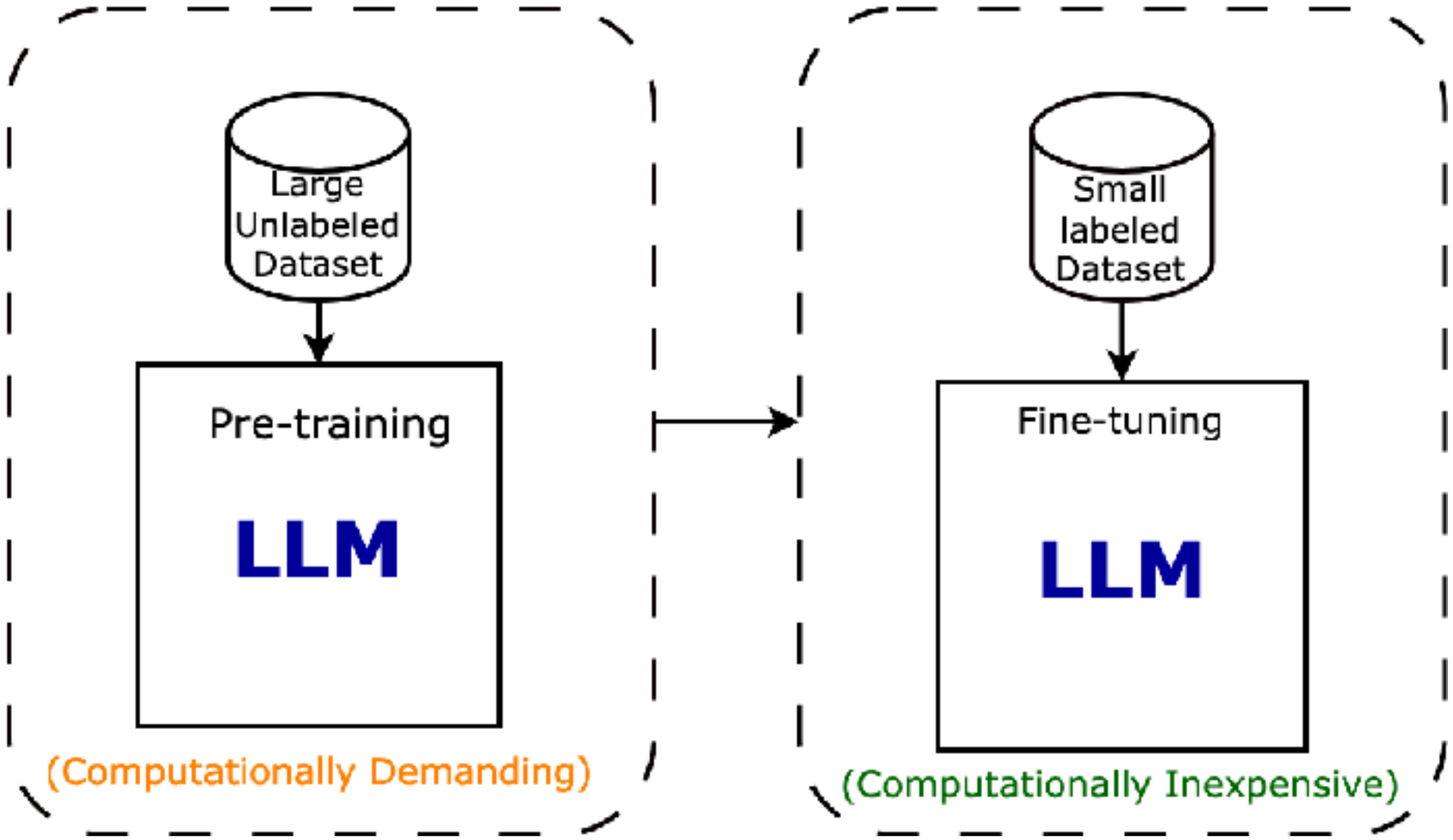
Flow Example



Fine-Tuning

RAG vs Fine-Tuning

Fine-Tuning



RAG vs Fine-tuning

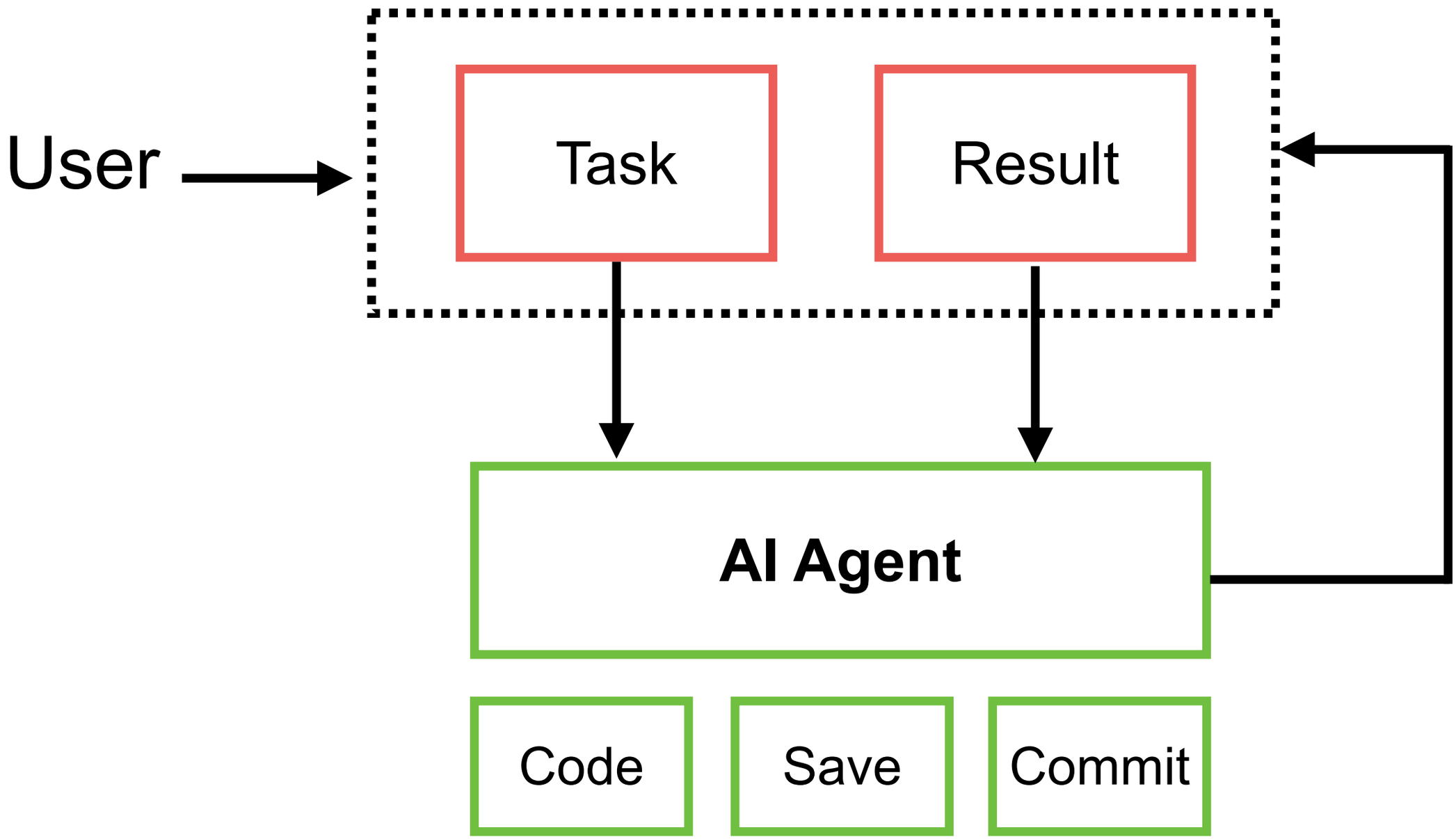
Scenario	RAG Preferred	Fine-Tuning Preferred
Skillset Requirements	Strong in information retrieval engineering	Strong in deep learning model fine-tuning
Data Freshness	Real-time or frequently updated data needed	Static, domain-specific data suffices
Domain Complexity	Multiple domains or high data diversity	Specialized, jargon-heavy domains (e.g., medical)
Resource Needs	Lower computation for diverse, dynamic responses	High computational power available for model tuning

https://www.linkedin.com/pulse/fine-tuning-vs-prompting-rag-which-pick-your-llm-dr-rabi-prasad-722cc/

https://www.kdnuggets.com/go-out-stay-in-rag-vs-fine-tuning

AI Agent

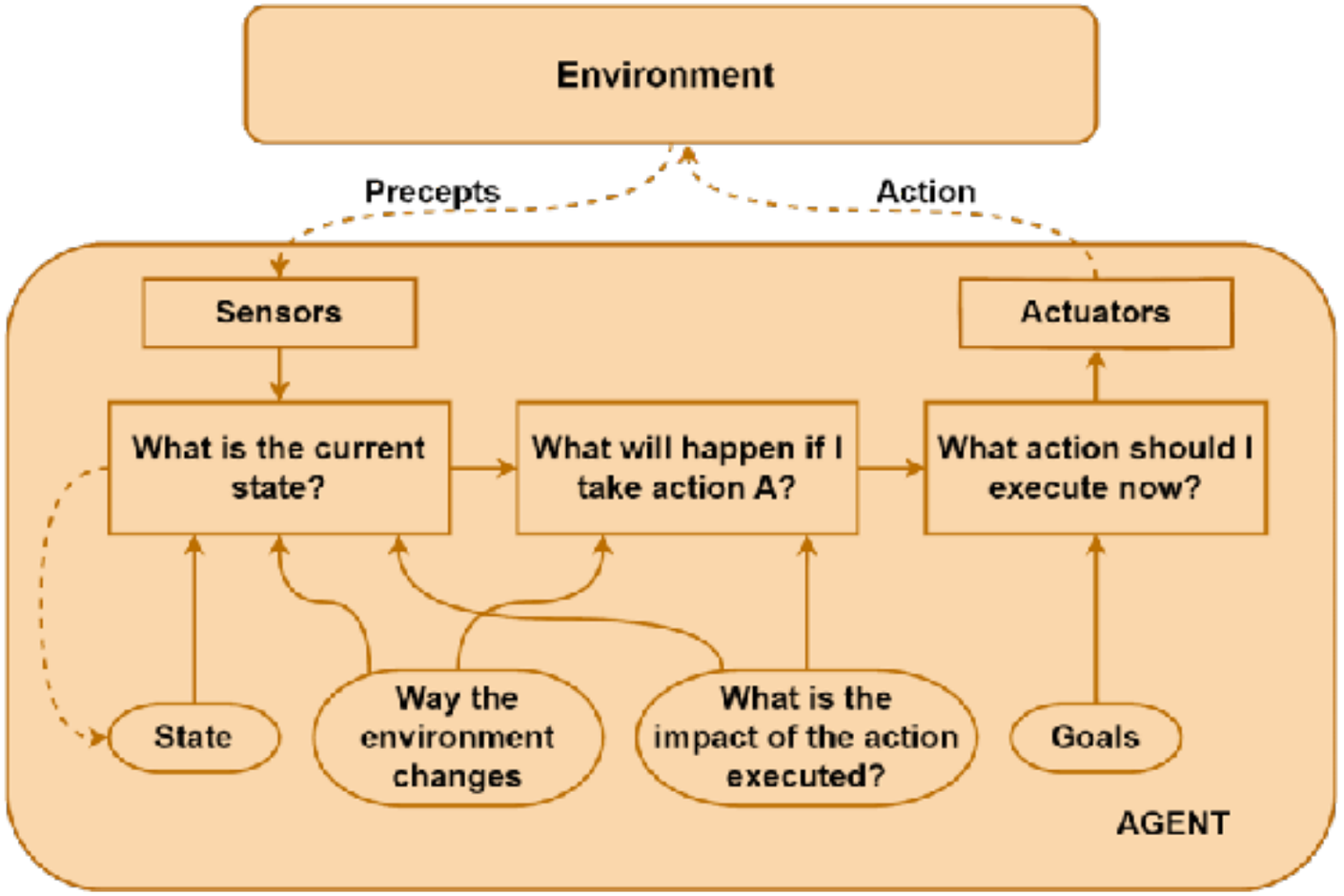
Goal or Result-based



https://github.com/e2b-dev/awesome-ai-agents

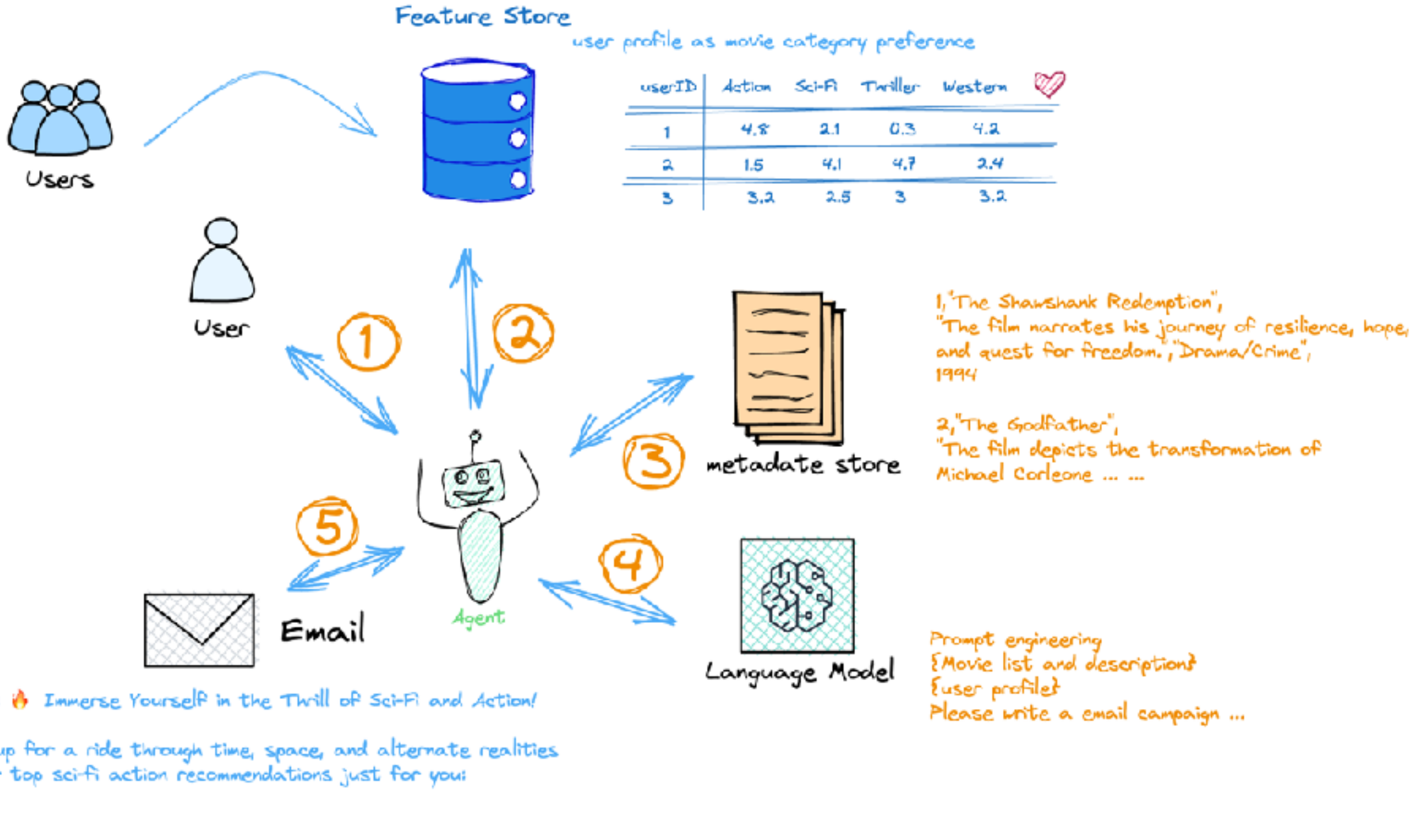
Goal-based agent

Attempts to choose best strategy to achieve goal on environment



https://www.baeldung.com/cs/goal-based-vs-utility-based-agents

AI Agent



https://aws.amazon.com/th/what-is/ai-agents/

Types of AI Agent

Simple reflex

Model-based reflex

Goal-based

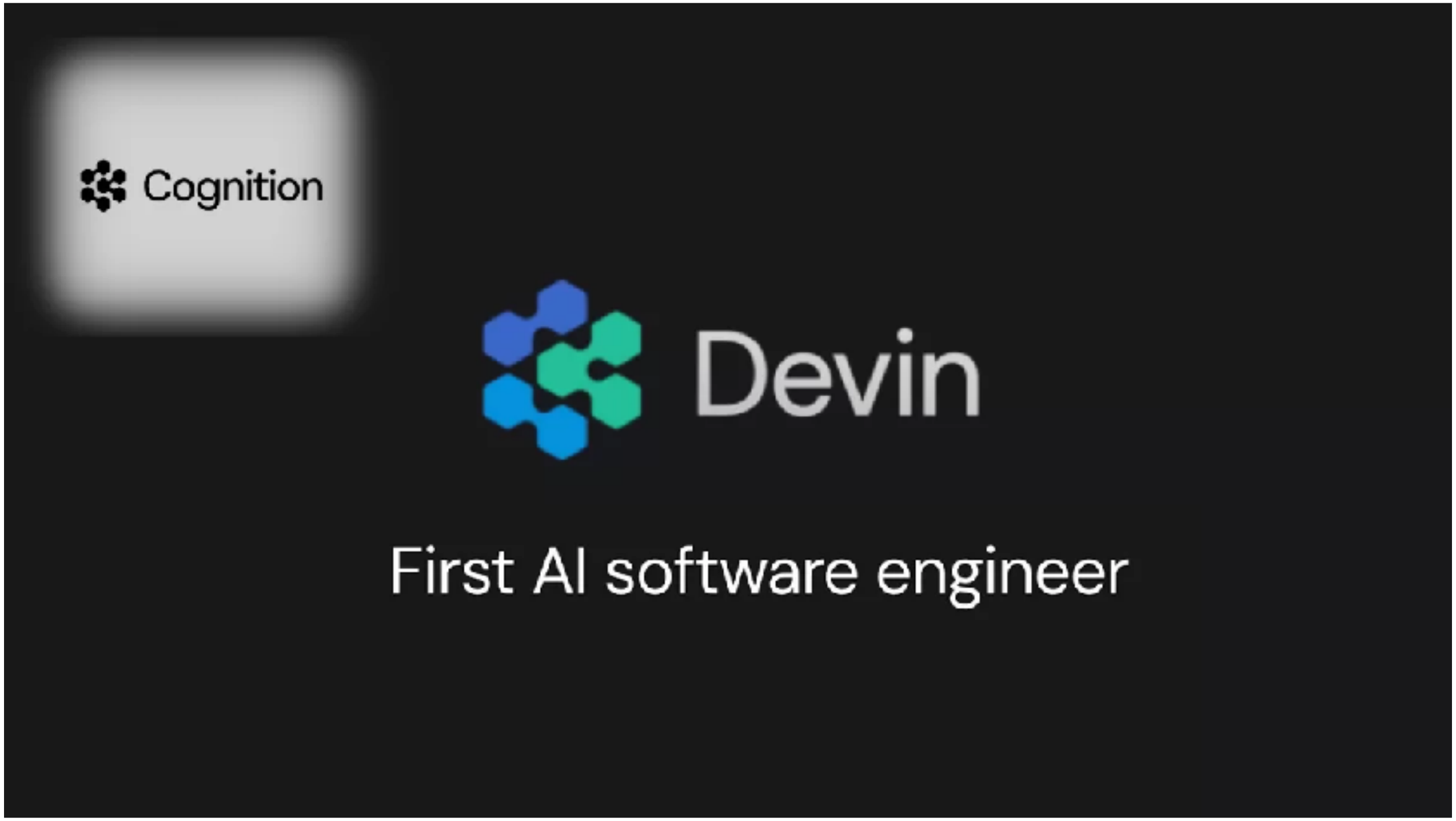
Utility-based

Learning

Hierarchical

https://github.com/e2b-dev/awesome-ai-agents

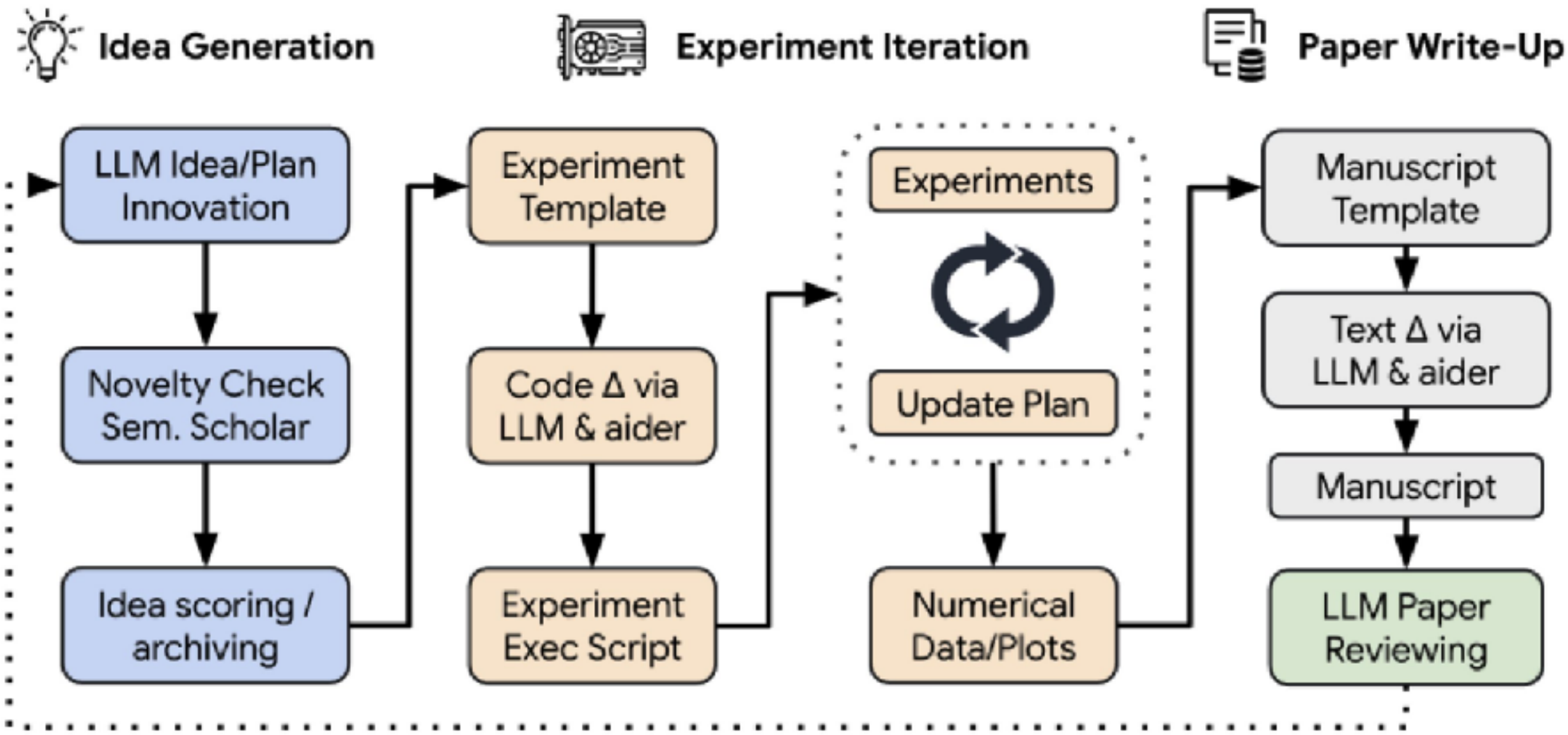
End-to-end Software Agent



https://www.cognition.ai/blog/introducing-devin

The AI Scientist: Towards Fully Automated Open-Ended Scientific Discovery

Chris Lu[1,2,*], **Cong Lu**[3,4,*], **Robert Tjarko Lange**[1,*], **Jakob Foerster**[2,†], **Jeff Clune**[3,4,5,†] **and David Ha**[1,†]
[*]Equal Contribution, [1]Sakana AI, [2]FLAIR, University of Oxford, [3]University of British Columbia, [4]Vector Institute, [5]Canada CIFAR AI Chair, [†]Equal Advising



https://arxiv.org/abs/2408.06292

Let’s Start

Chat and Search

Chat and Search

Gemini

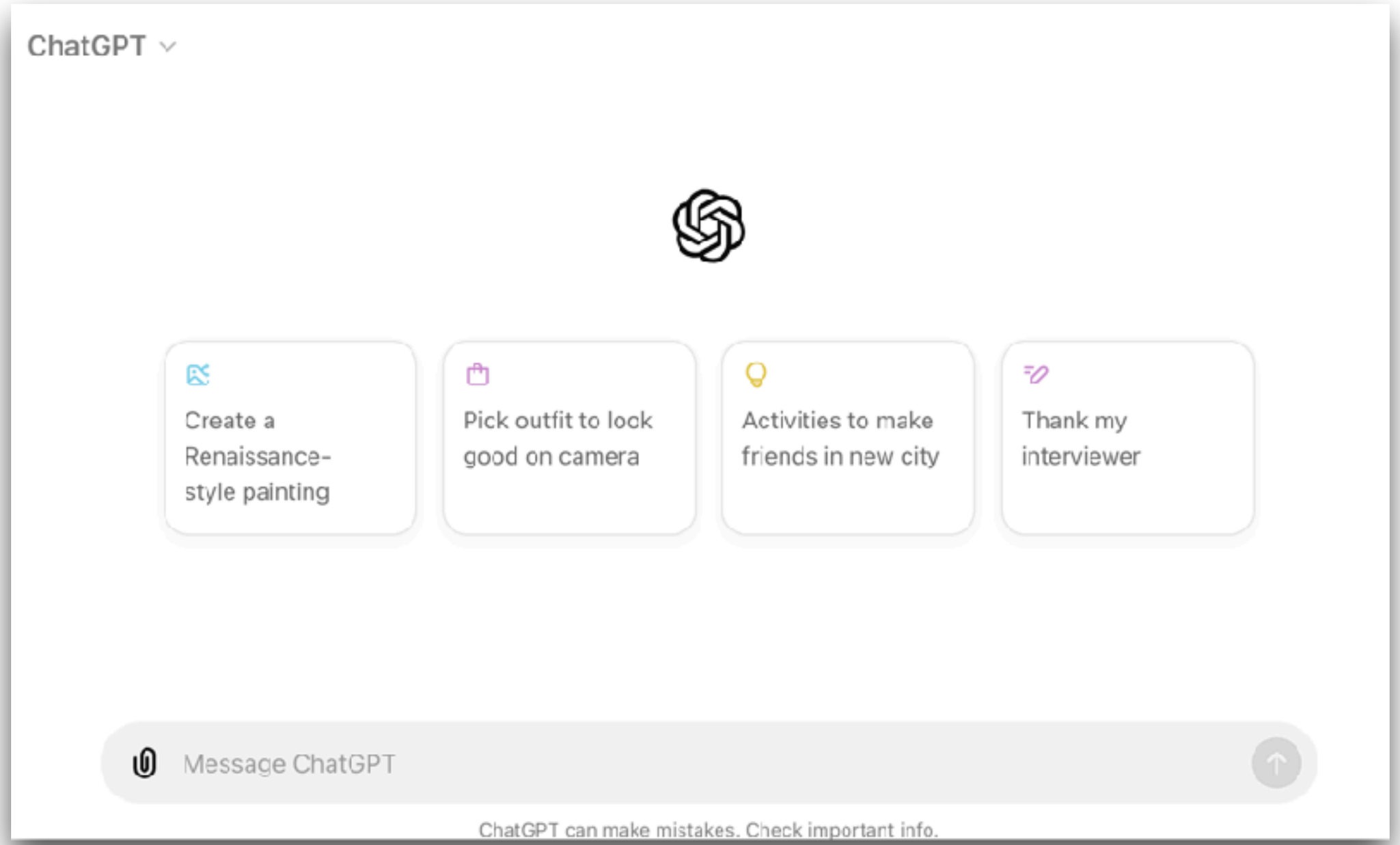
Claude

ChatGPT

Pi

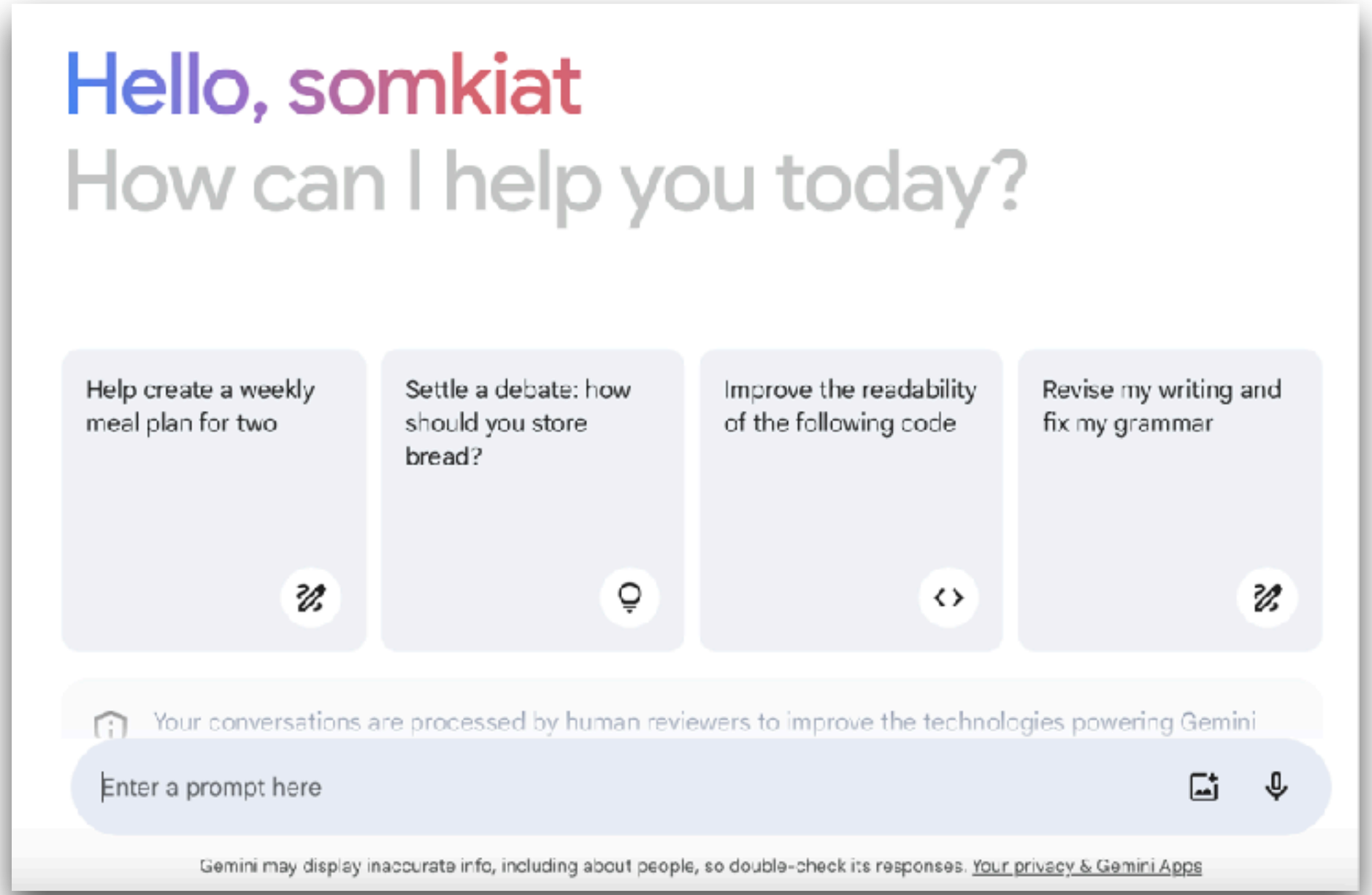
perplexity

ChatGPT from OpenAI



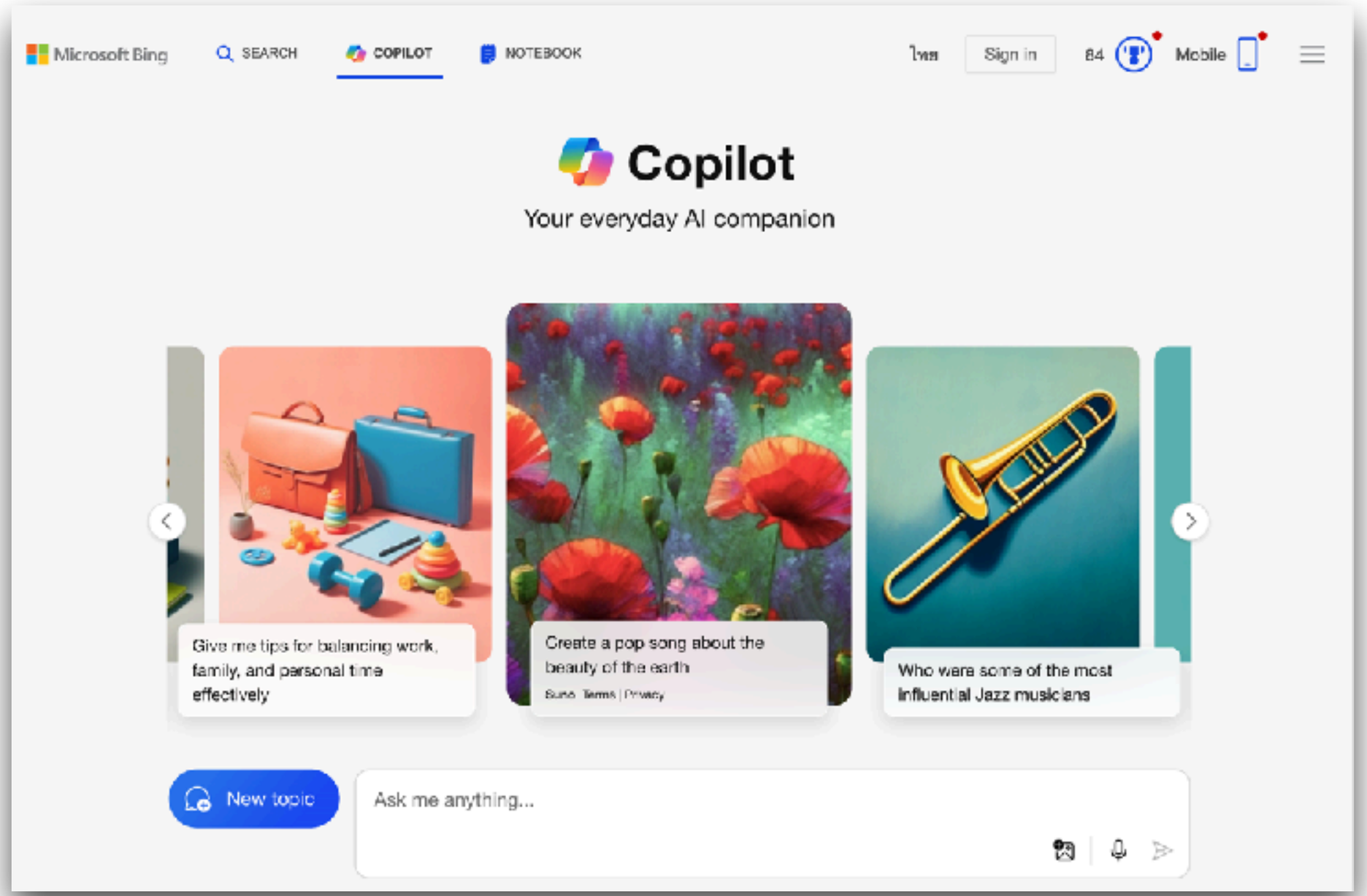
https://chatgpt.com/

Google Gemini



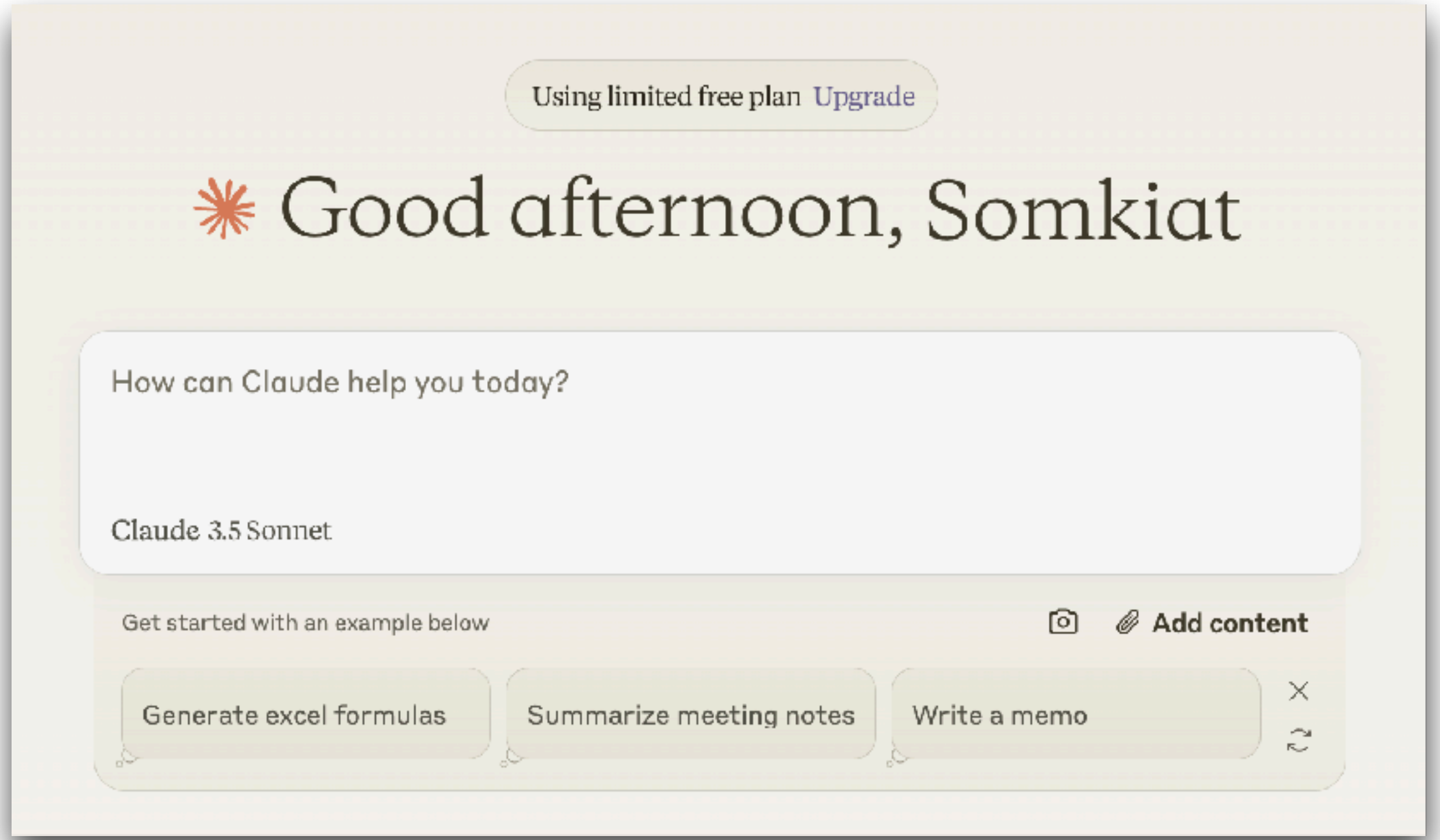
https://gemini.google.com/app

Microsoft Bing Copilot



https://www.bing.com/chat

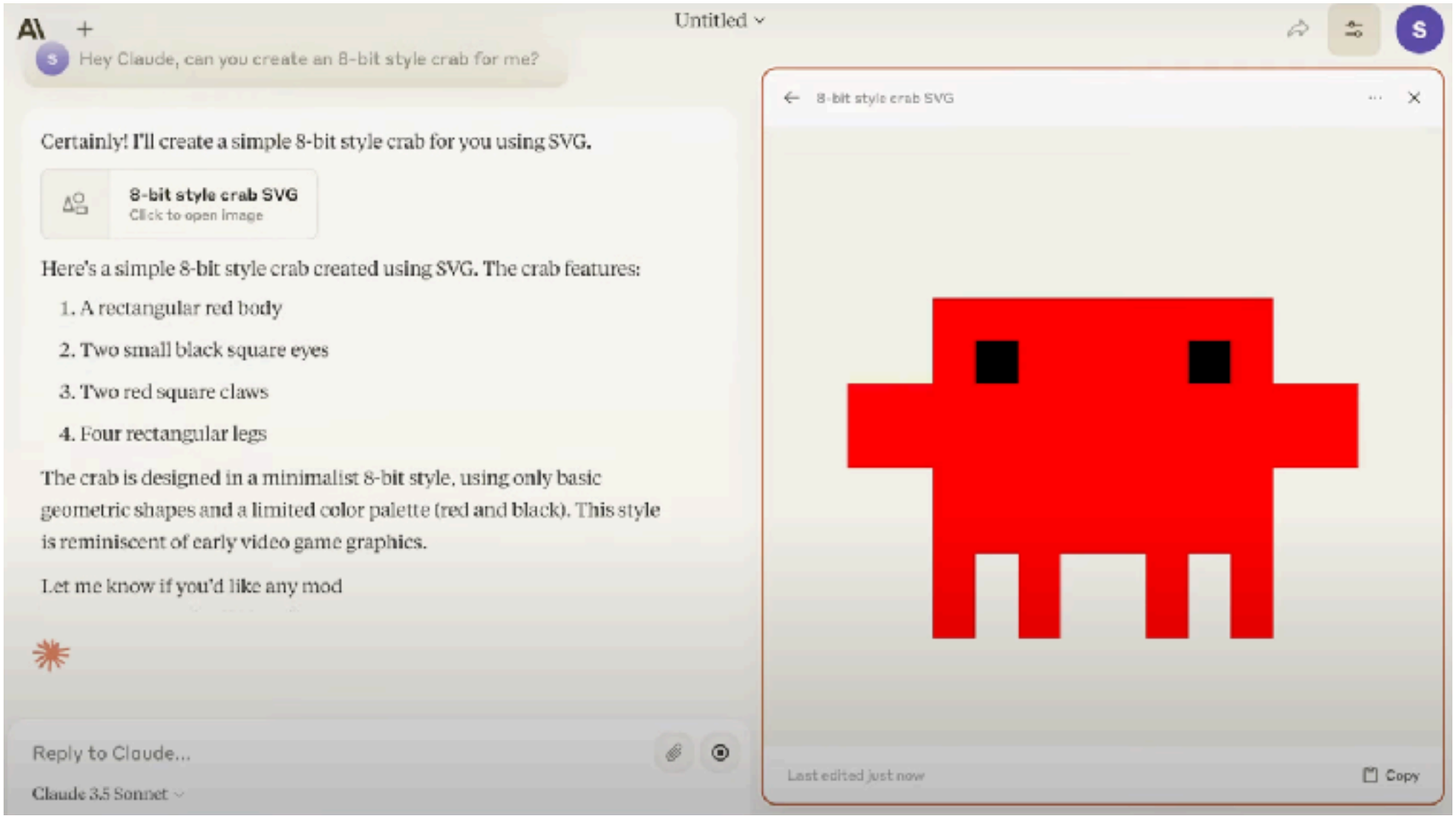
Claude.AI



https://claude.ai/new

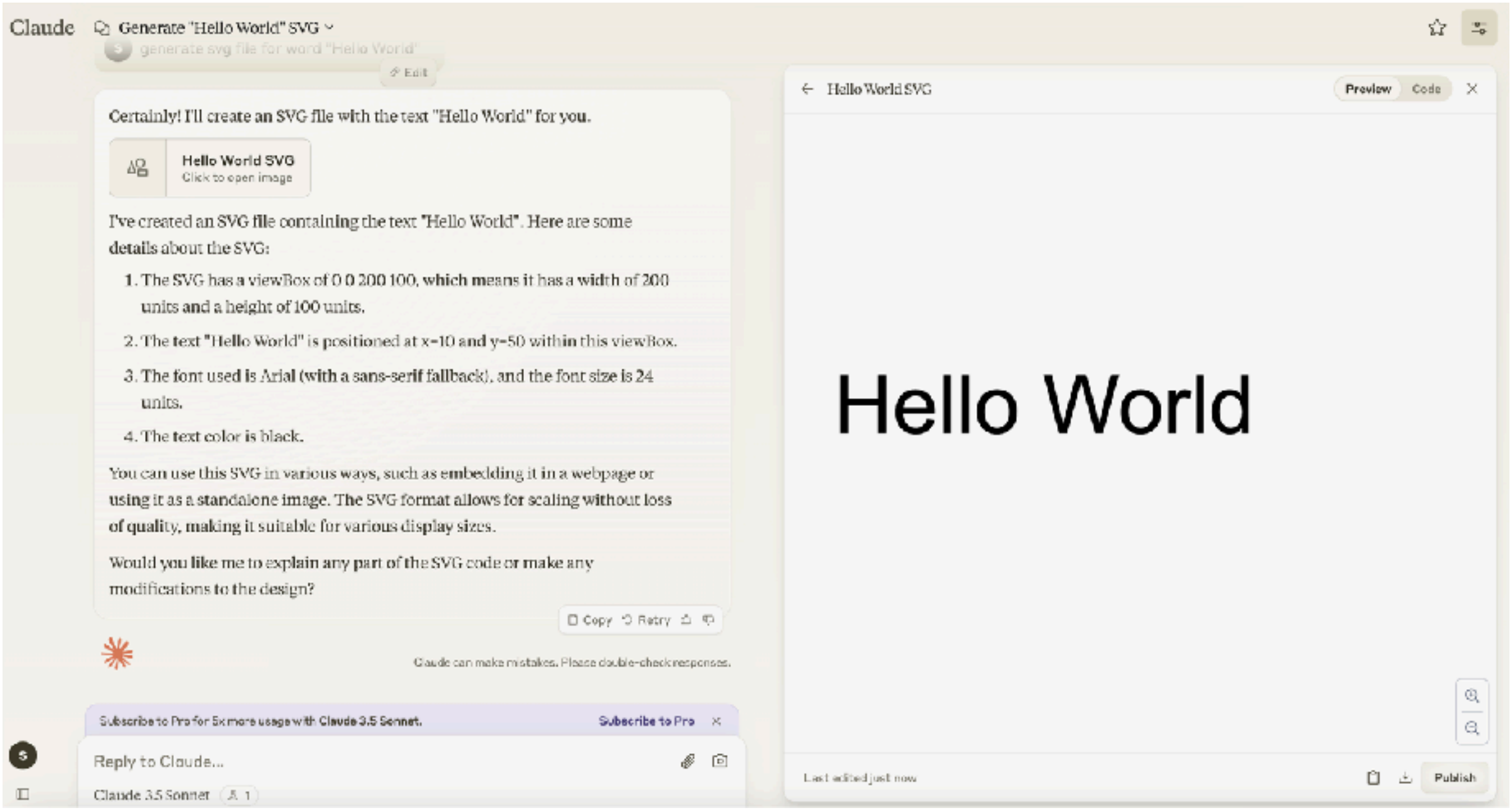
Claude AI + Artifact

Try to automatically compile and run generated code



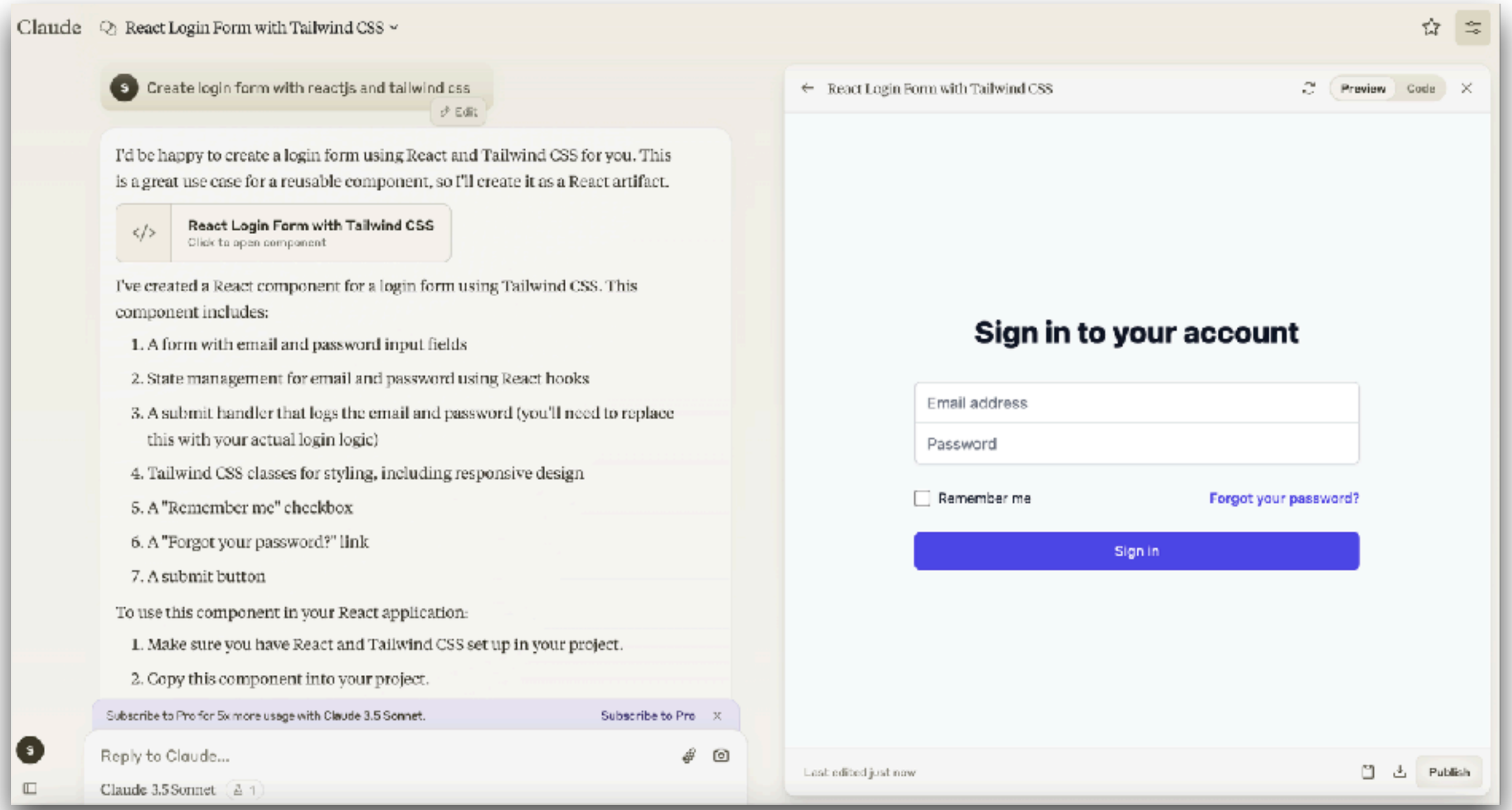
https://www.anthropic.com/news/claude-3-5-sonnet

Demo



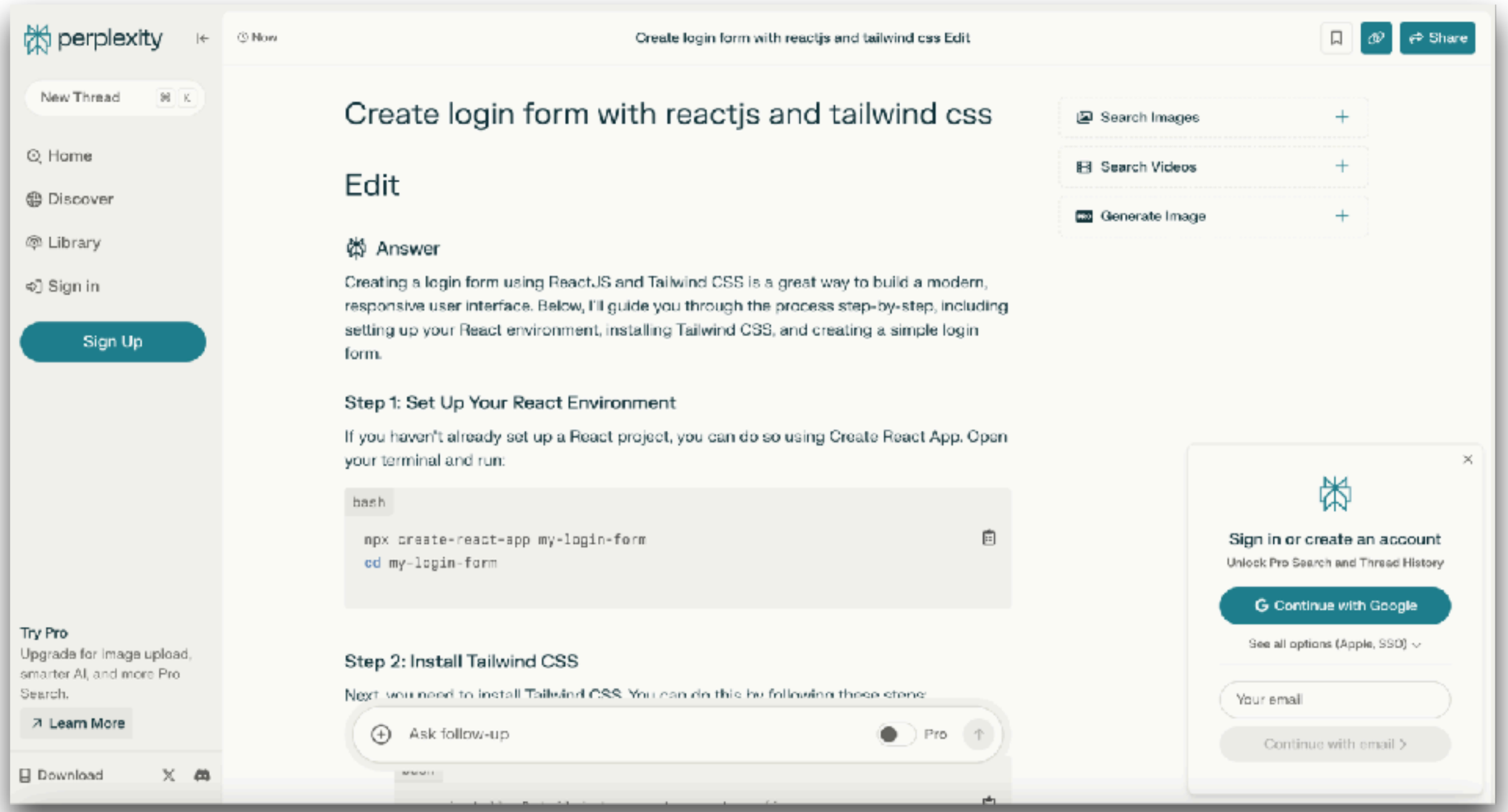
https://claude.ai/

Demo



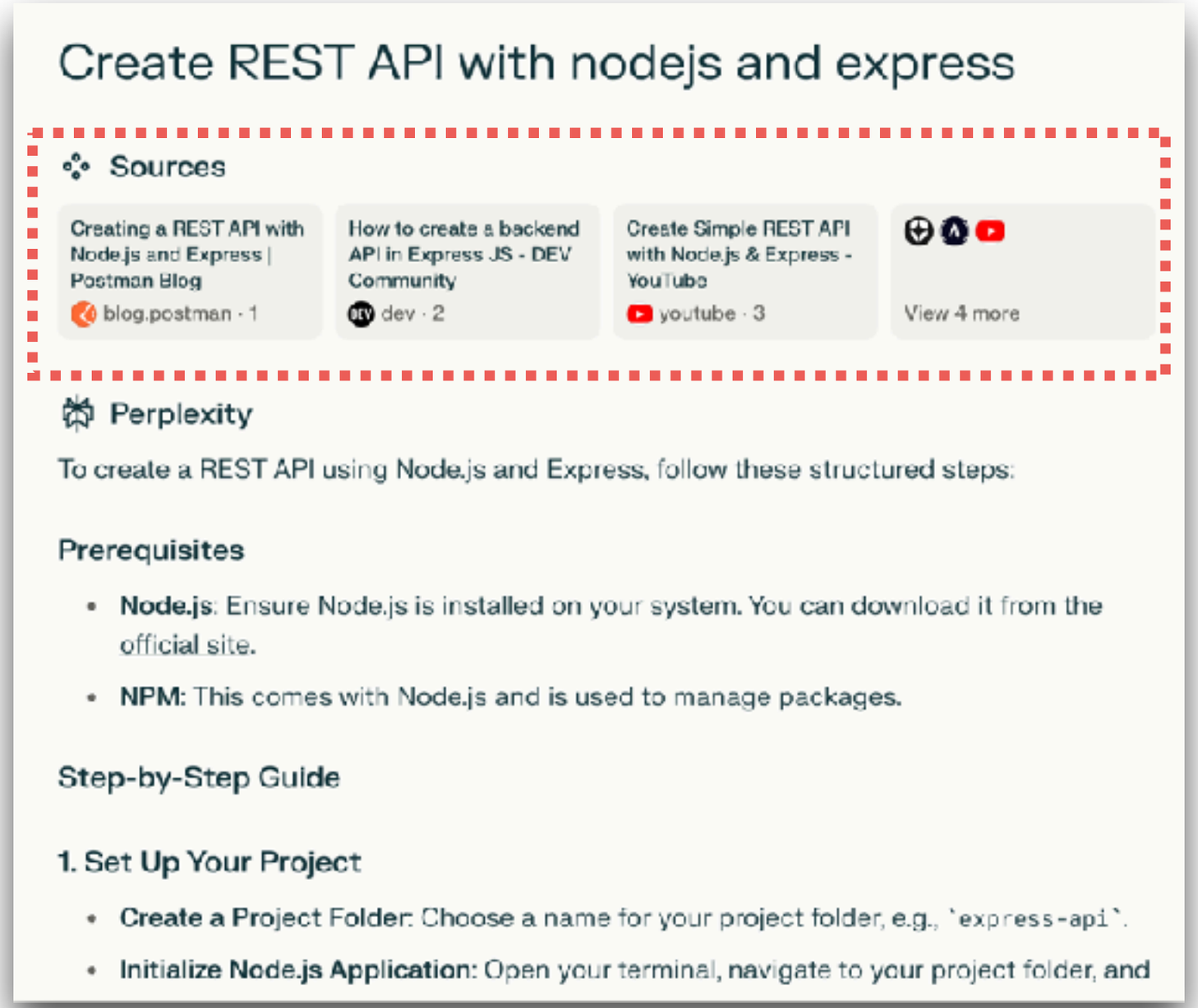
https://claude.ai/

Perplexity.ai



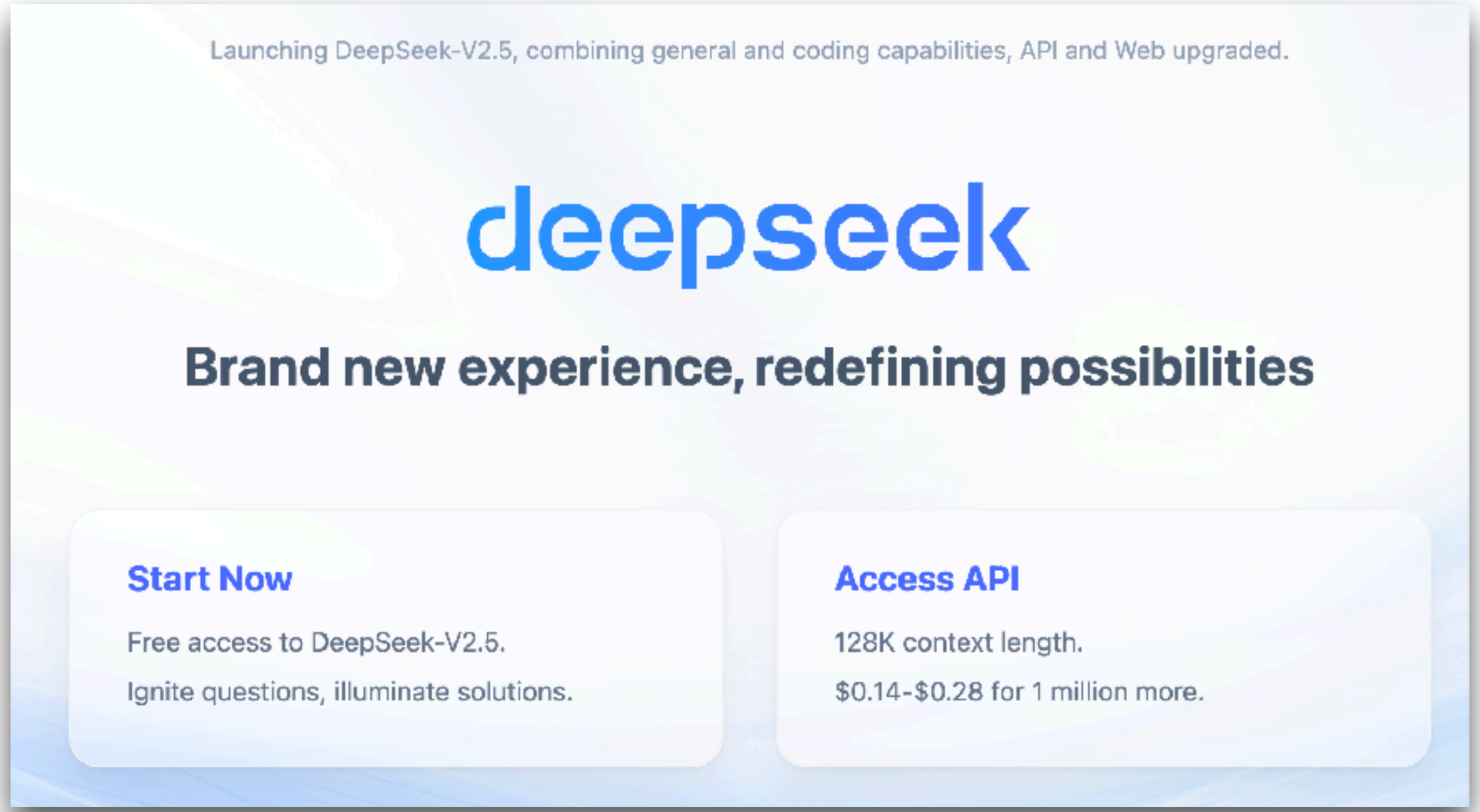
https://www.perplexity.ai/

Perplexity.AI



https://www.perplexity.ai/search/create-rest-api-with-nodejs-an-H1bLD2YdSJyQgOrjbgvk5g

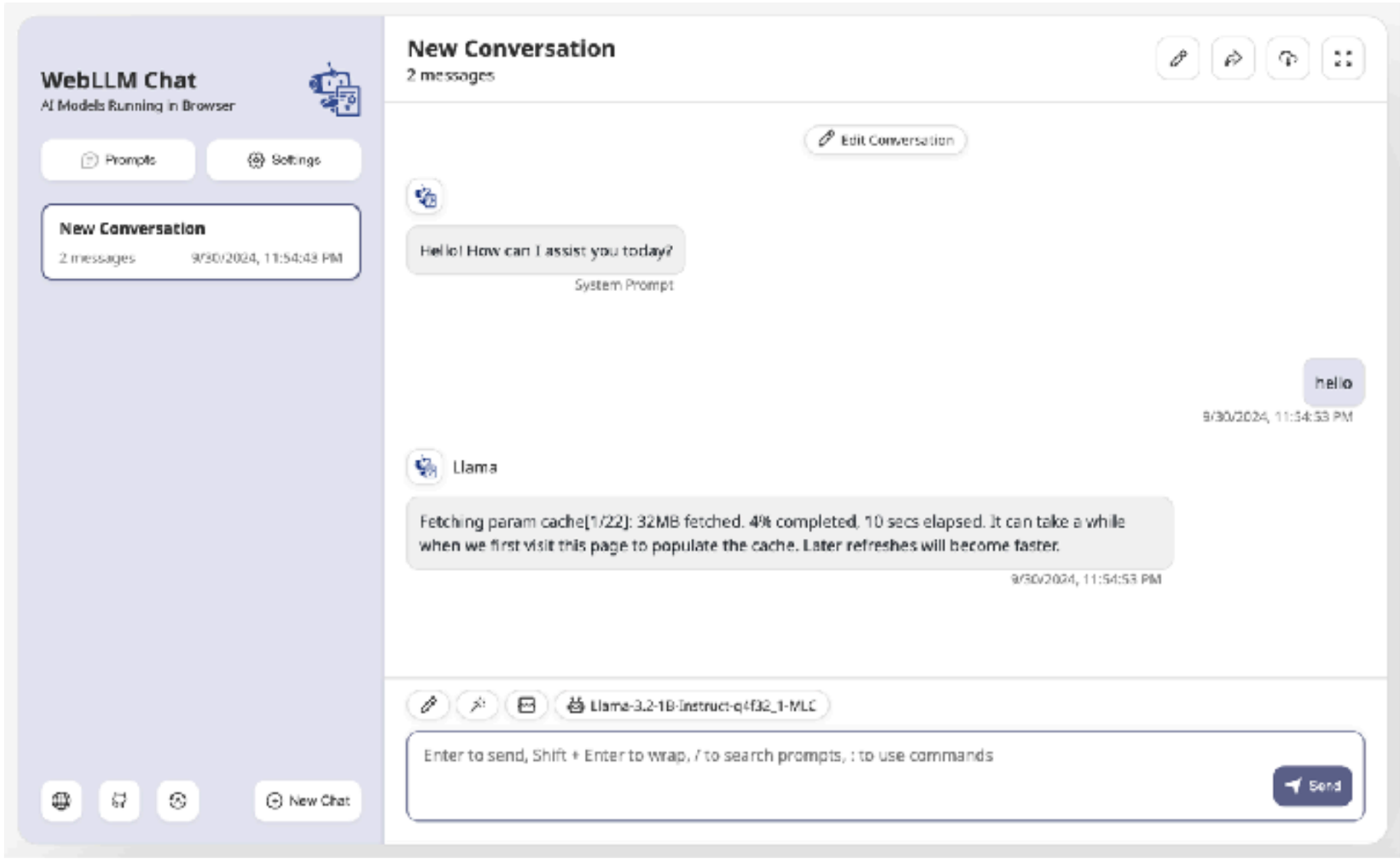
DeepSeek



https://www.deepseek.com/

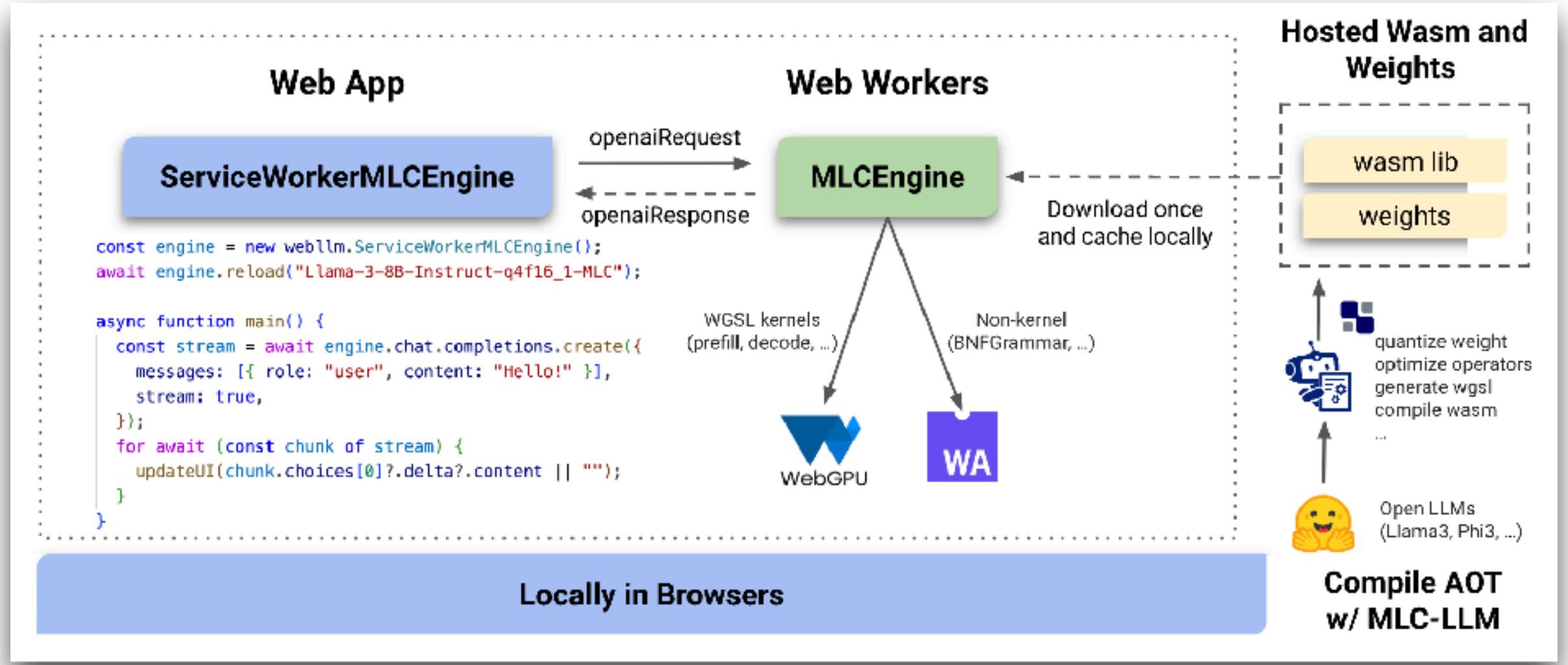
WebLLM Chat

High-performance in-browser LLM inference engine



https://webllm.mlc.ai/

WebLLM Architecture



https://blog.mlc.ai/2024/06/13/webllm-a-high-performance-in-browser-llm-inference-engine

SearchGPT

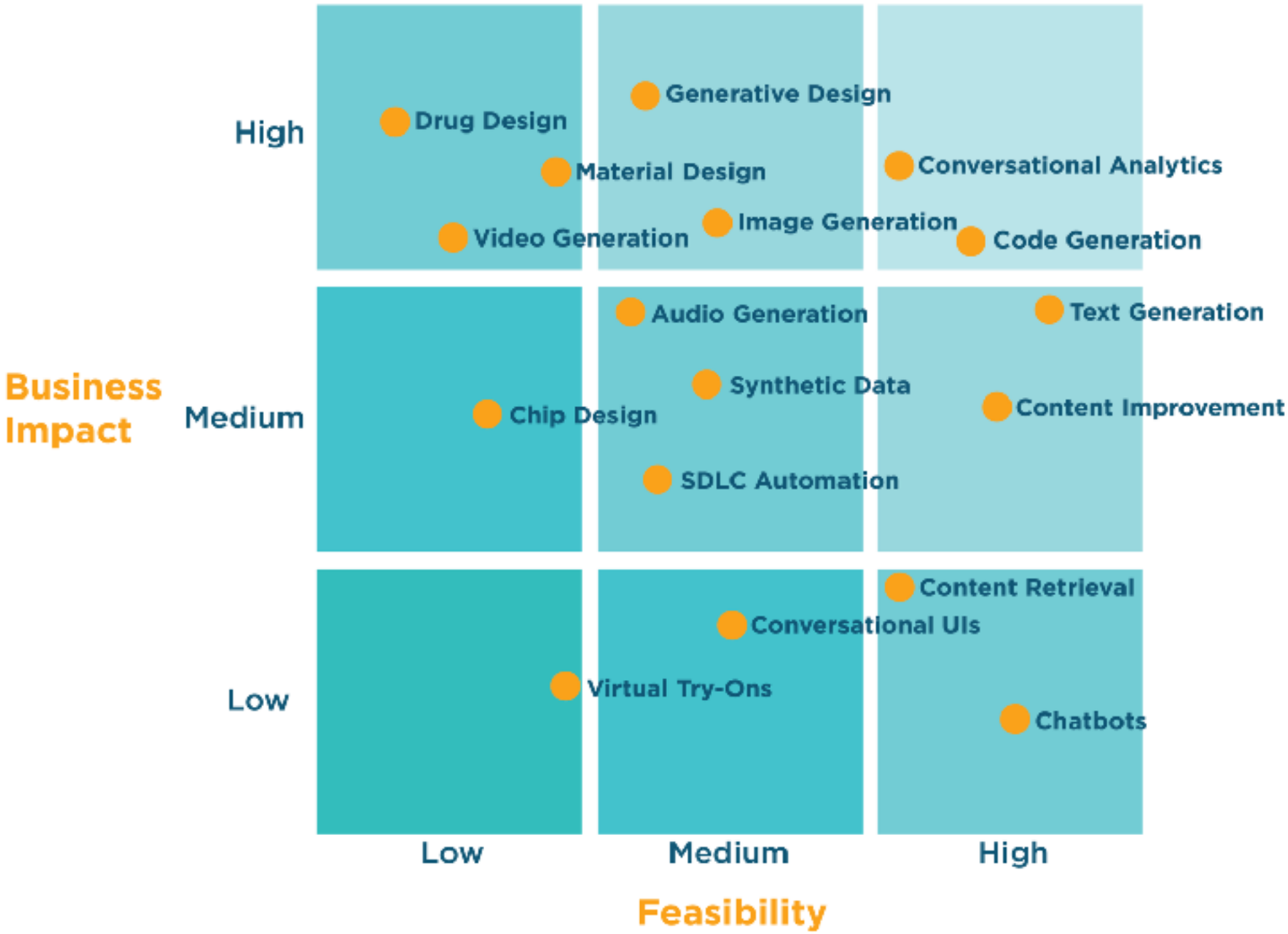
https://openai.com/index/searchgpt-prototype/

Chat and Search

User	Company	Console	Best model
ChatGPT	OpenAI	OpenAI platform	GPT-4o
Claude	Anthropic	Anthropic console	Claude 3.5 sonnet
Gemini	Google	Google AI Studio	Gemini 1.5 pro

Generative AI Use Cases

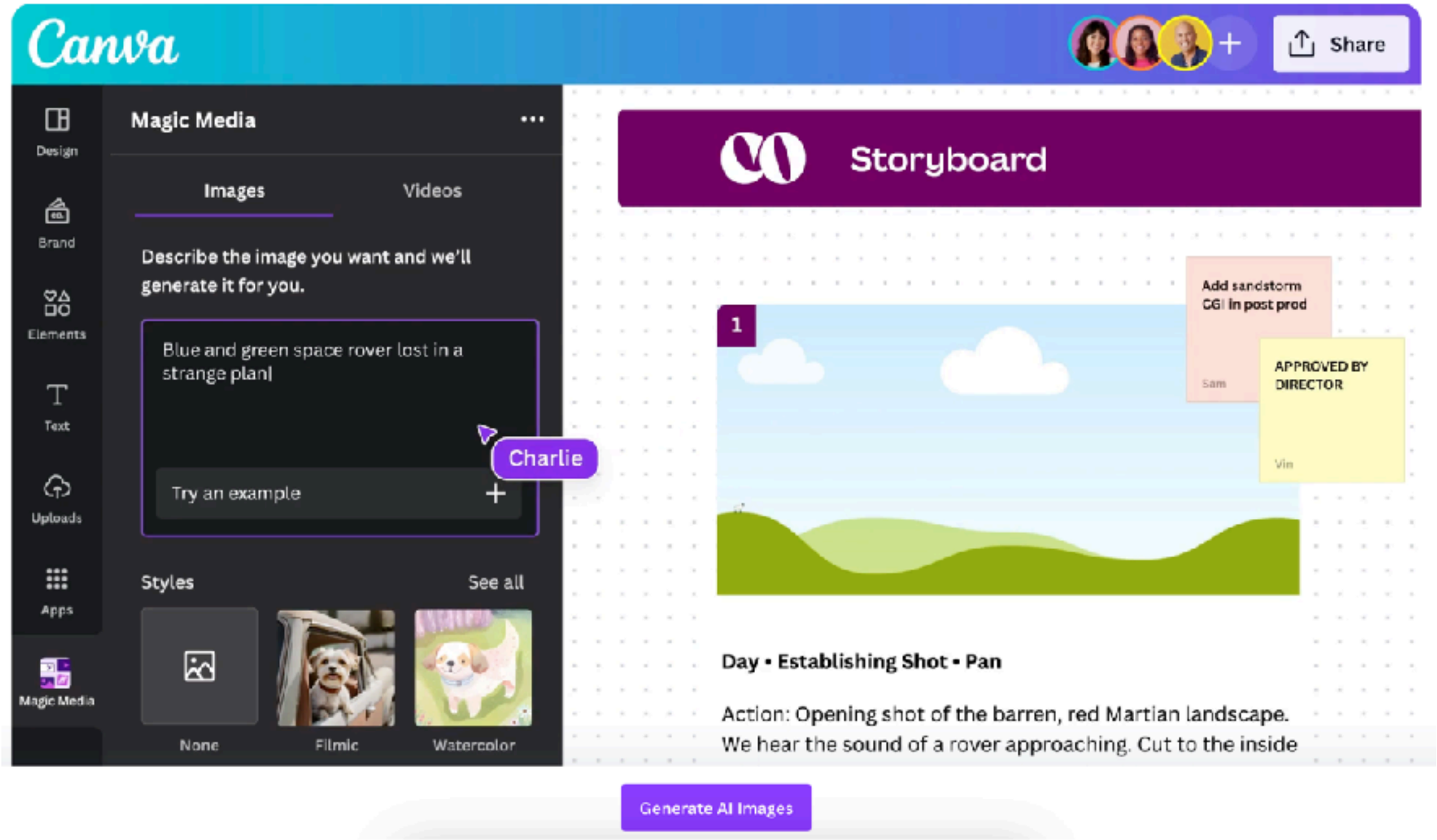
Generative AI Use Case Impact/Feasibility Matrix



https://altair.com/blog/executive-insights/want-to-identify-good-generative-ai-use-cases-dont-be-boring

Generative Design

Canva



https://www.canva.com/ai-image-generator/

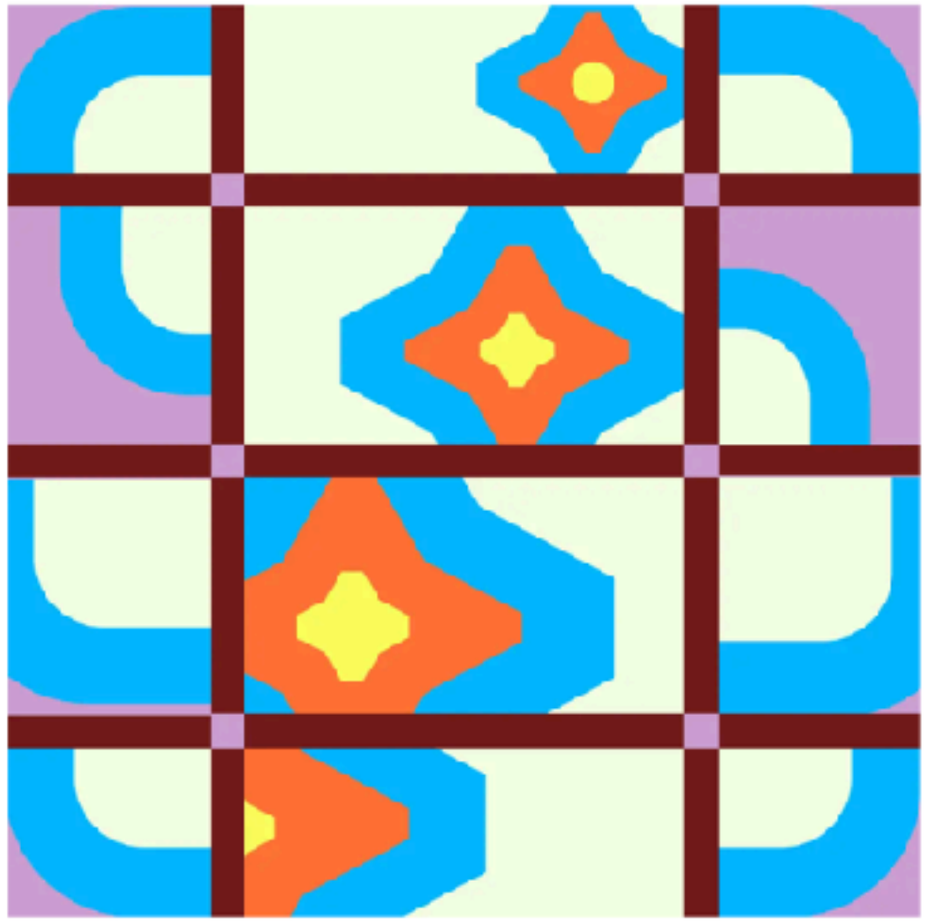
Figma AI

Products Solutions Community Resources Pricing Contact sales Log in Get started for free

Your creativity, unblocked with Figma AI

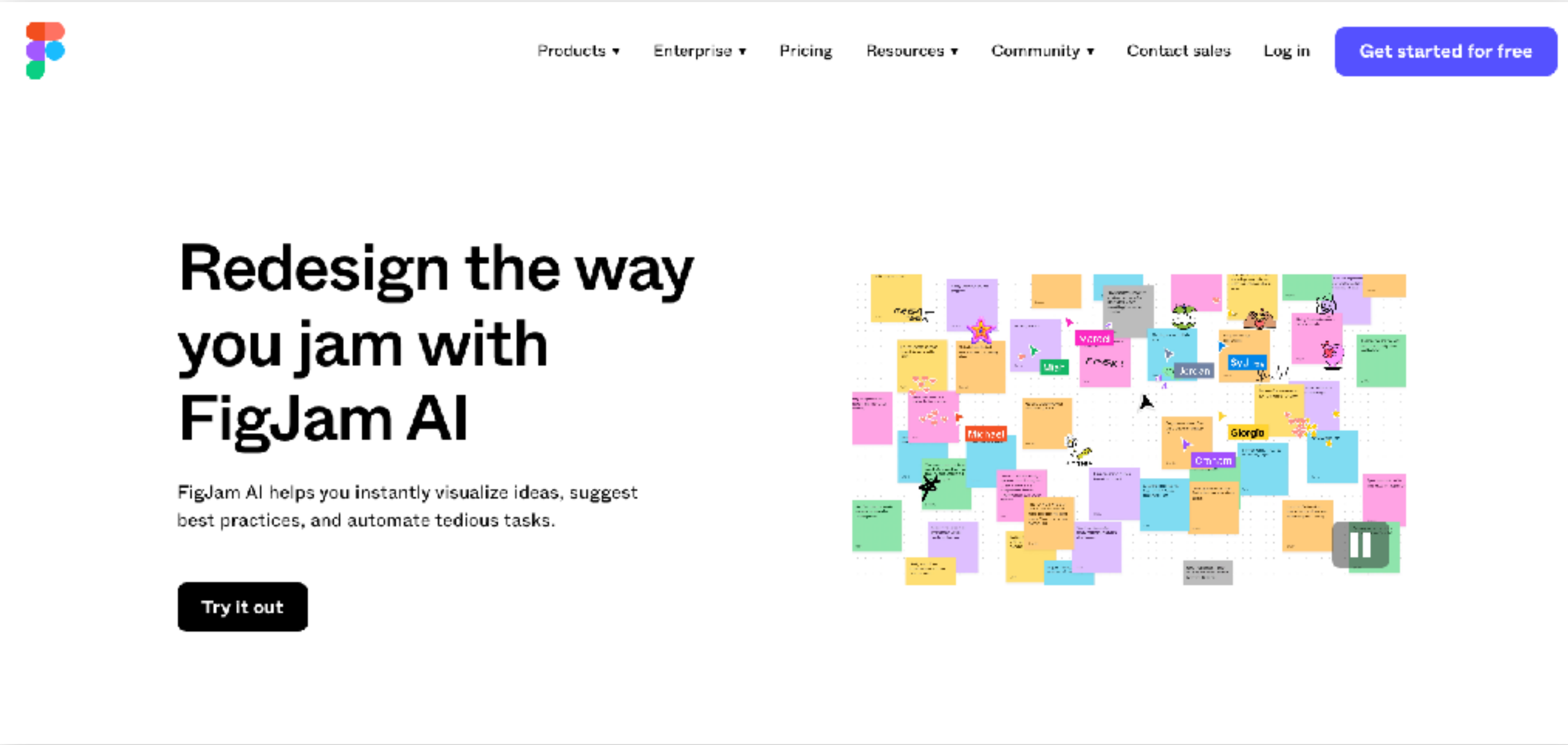
Get started faster, find what you're looking for, and stay in the flow. Make space for more creativity.

Learn about our approach to building AI



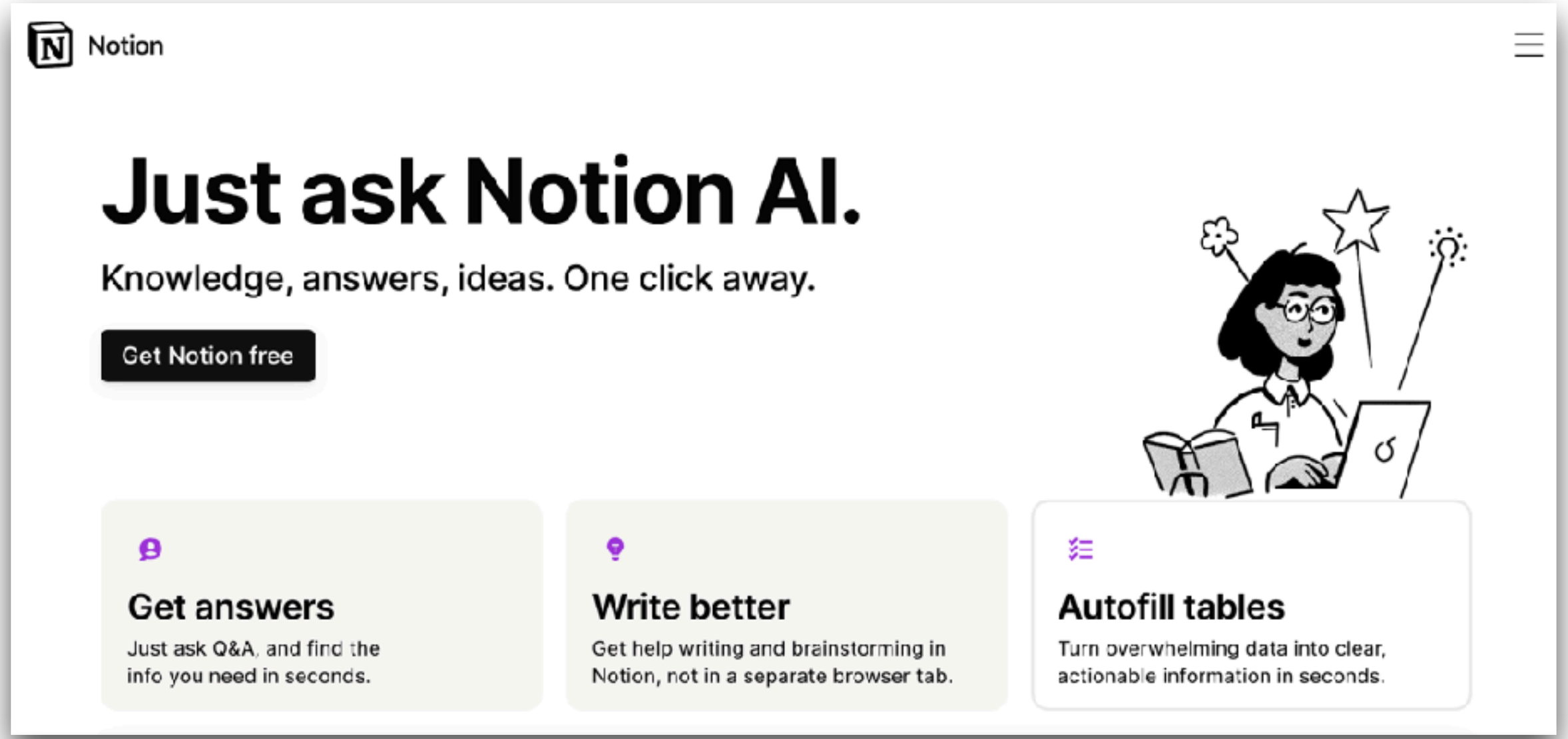
https://www.figma.com/ai/

FigJam AI



https://www.figma.com/figjam/ai/

Notion AI



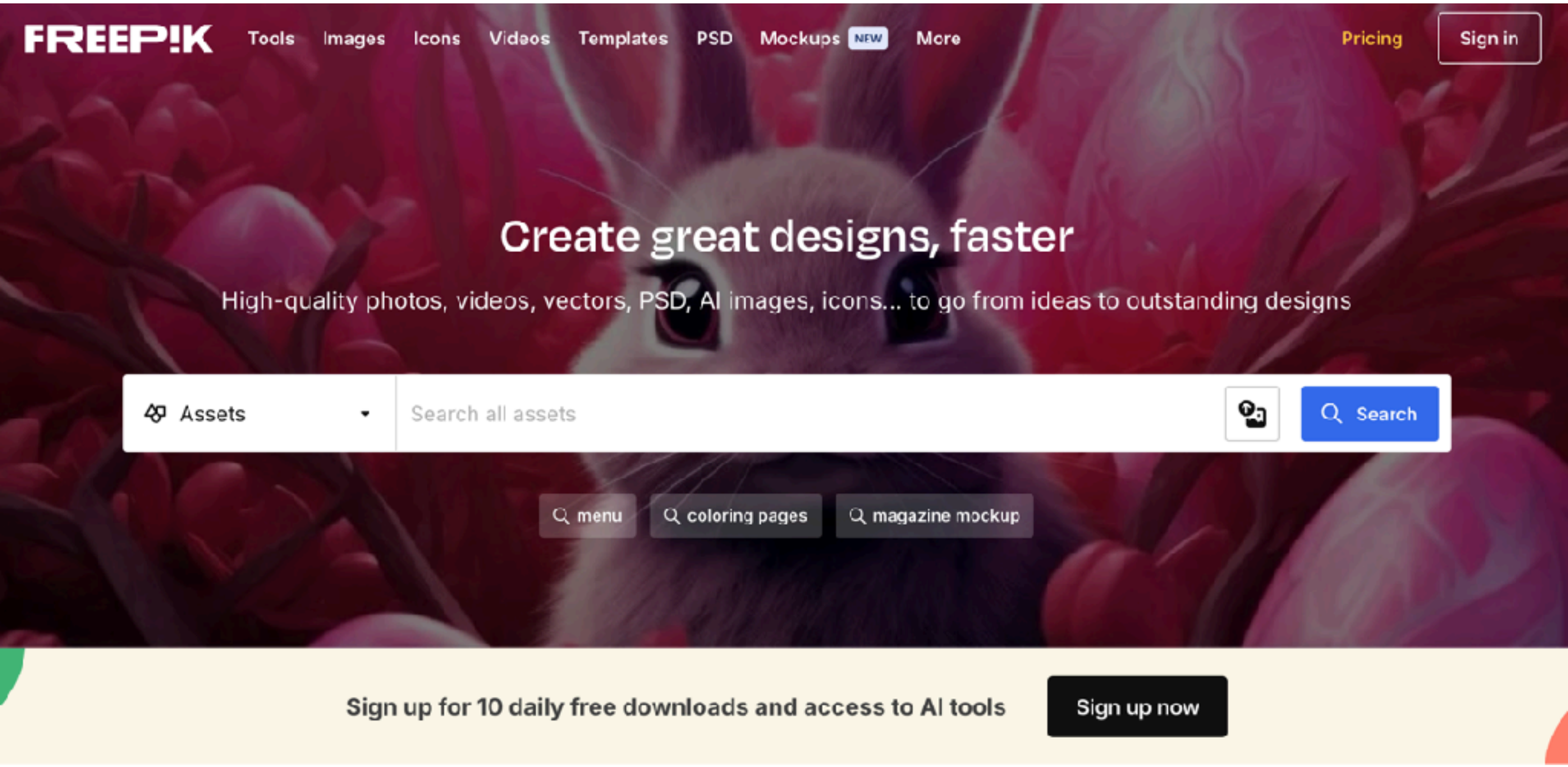
https://www.notion.so/product/ai

Generative Generation

Mid Journey

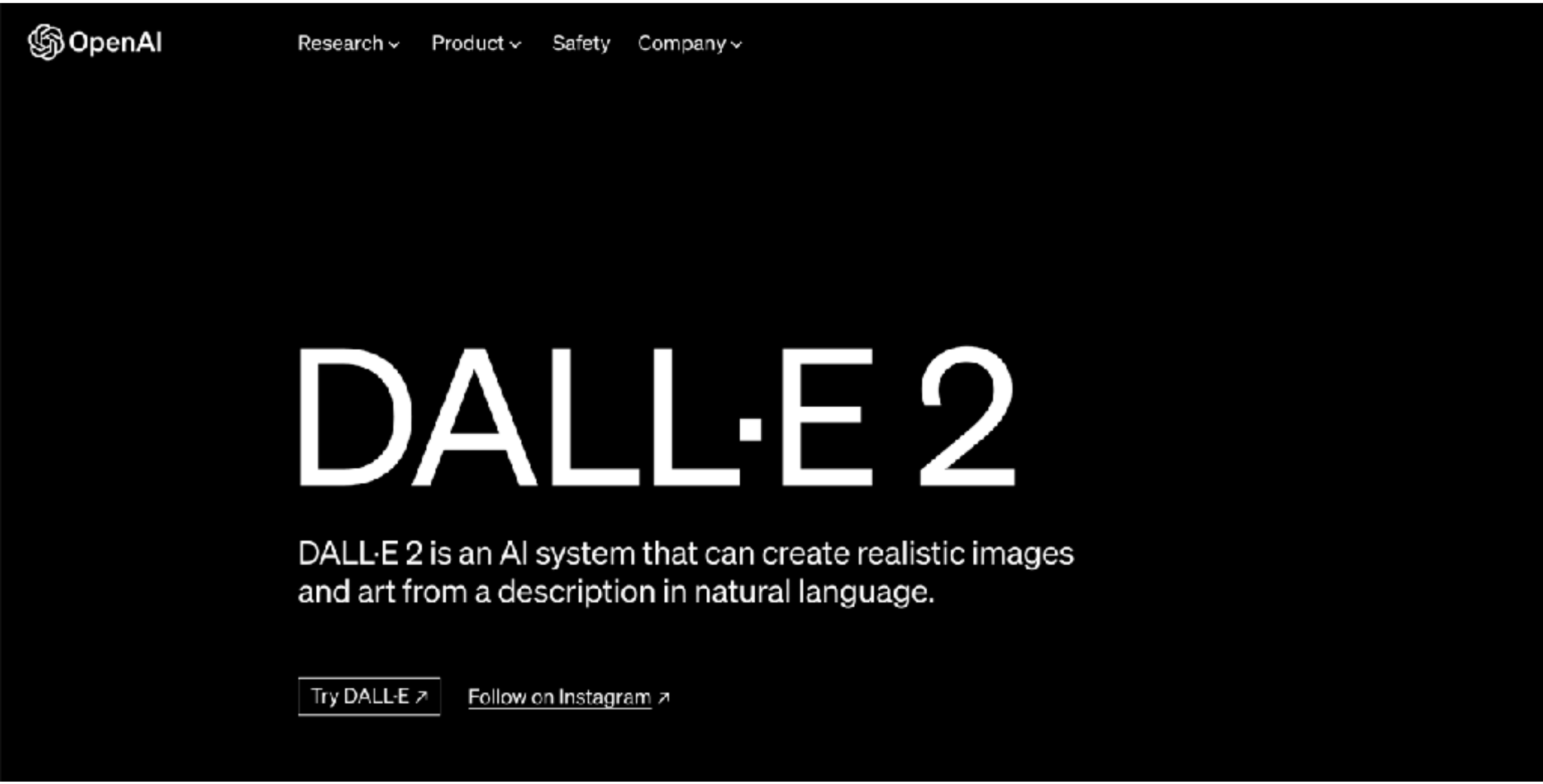
https://www.midjourney.com/

FreePik



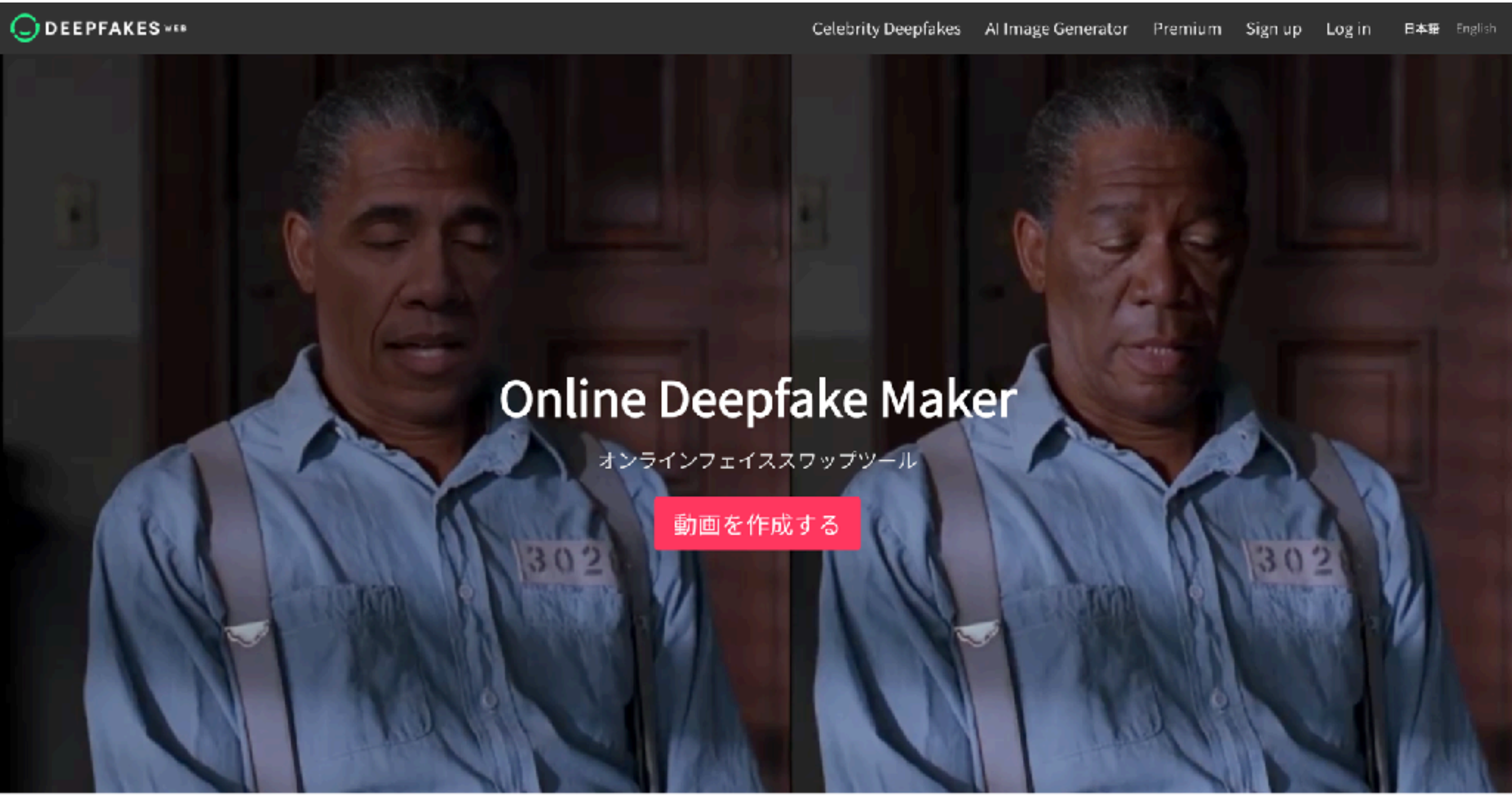
https://www.freepik.com/

DALL.E 2 from OpenAI



https://openai.com/dall-e-2

AI DeepFake !!



https://deepfakesweb.com/

Software Development Life Cycle

SDLC

Planning

- Written planning process
- Arch diagrams

Testing

- Automated testing
- TDD
- Testing environments
- Testing in prod
- Performance testing
- Load testing
- Generate test data

Development

- Automated dev env
- CI/CD
- Prototyping
- Code review
- Code generation
- Templates
- Cross-platform dev
- Preview env
- Post-commit code review
- Linting
- Static code analysis
- Project mgmt

Shipping

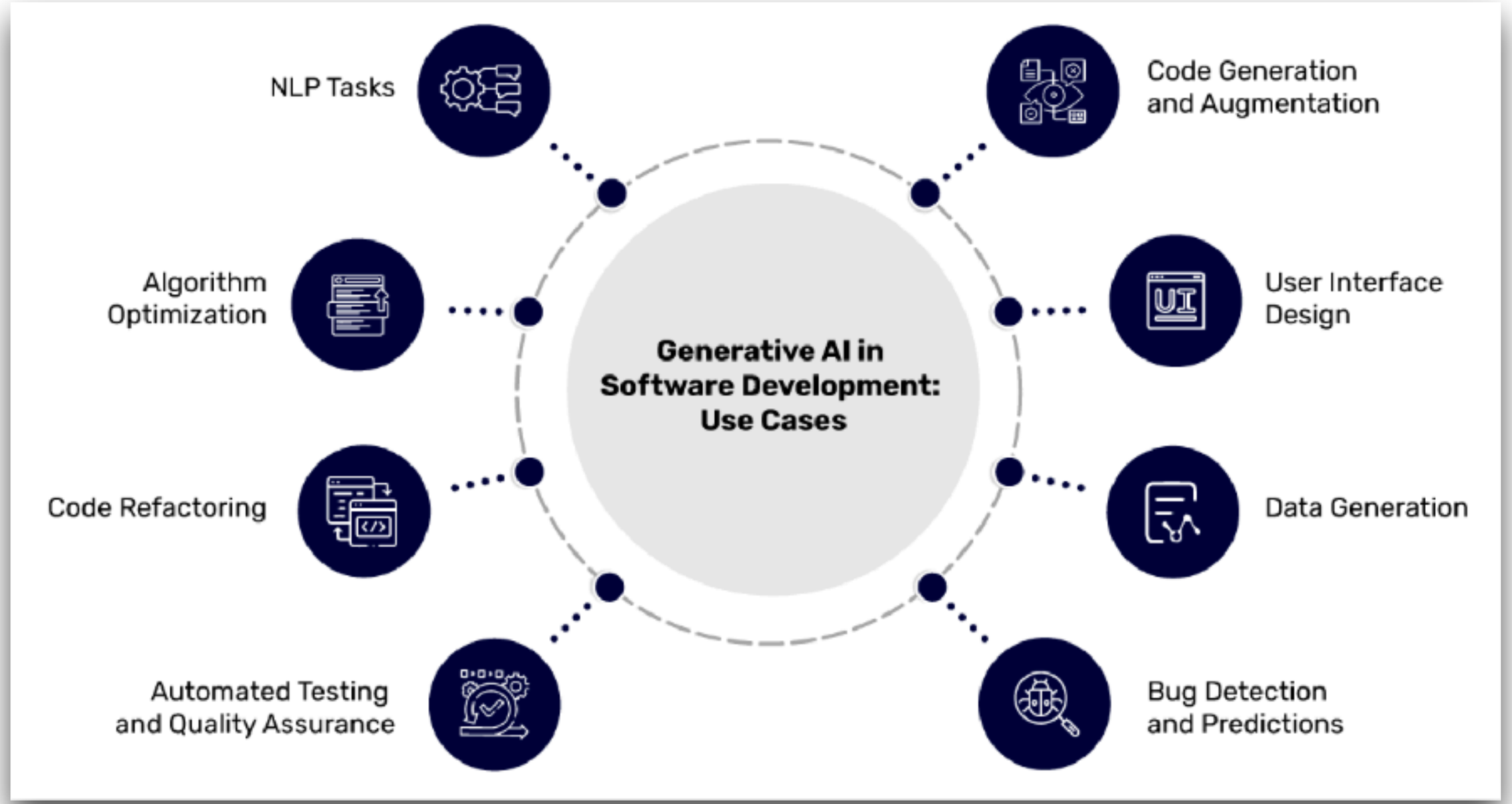
- FF & experimentation
- Logging
- Monitoring & alerting
- Staged rollouts

Maintenance

- Debug production
- Documentation
- Runbook examples
- Mitigation runbook

pragmaticengineer.com

SDLC



Impacts with productivity ?

Automated simple tasks

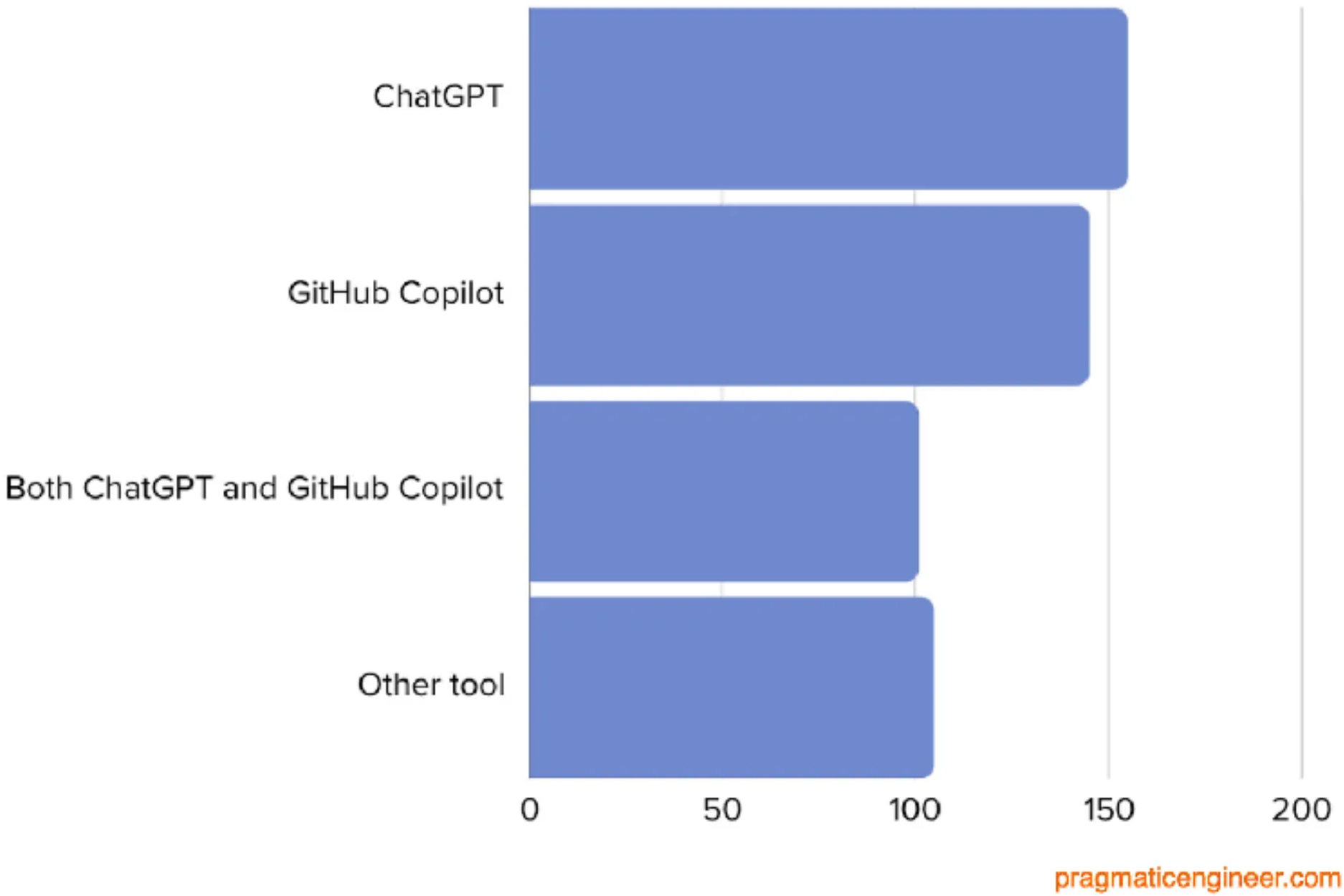
Improve quality and reliability

Improve communication

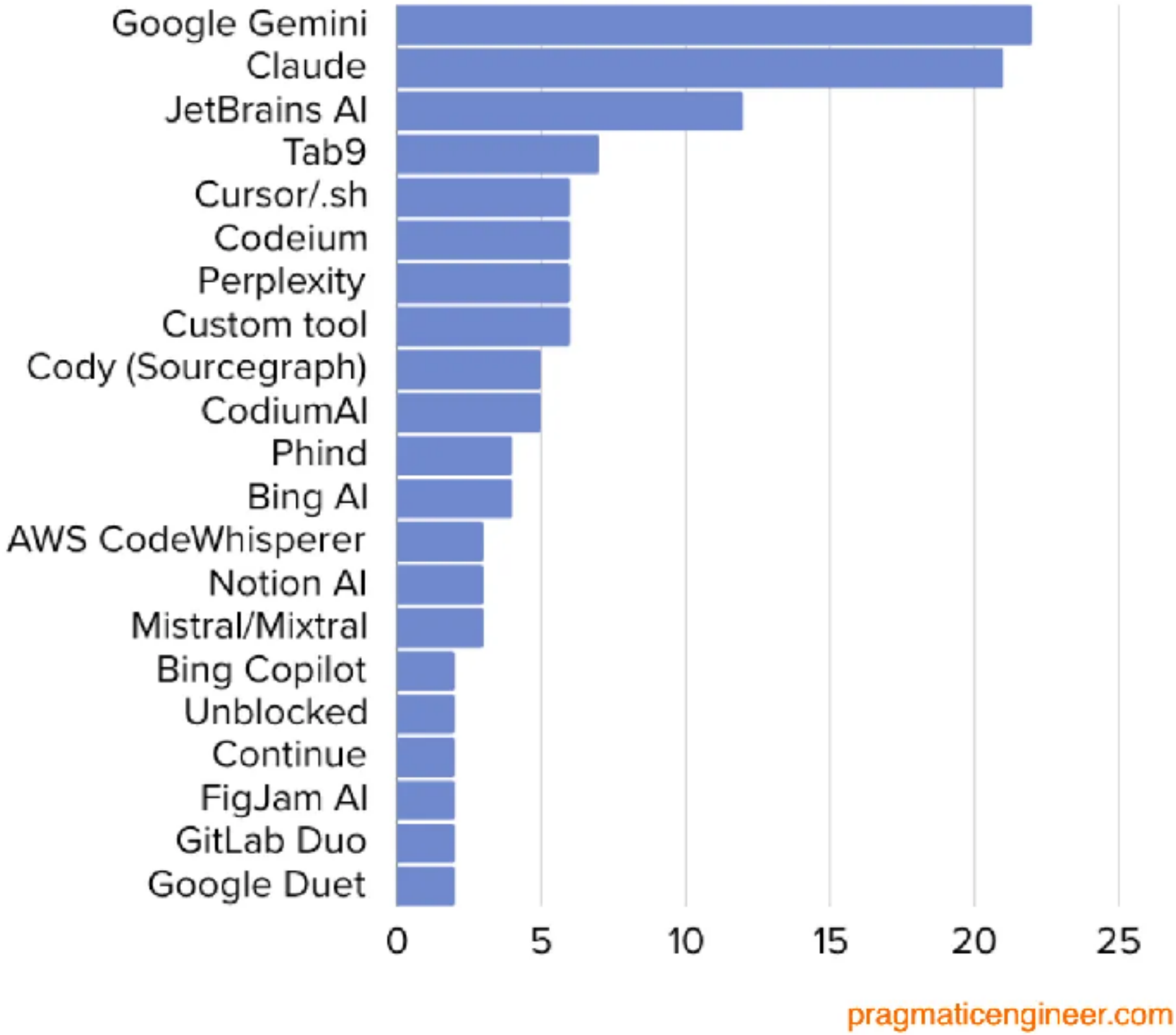
Faster prototype

Survey

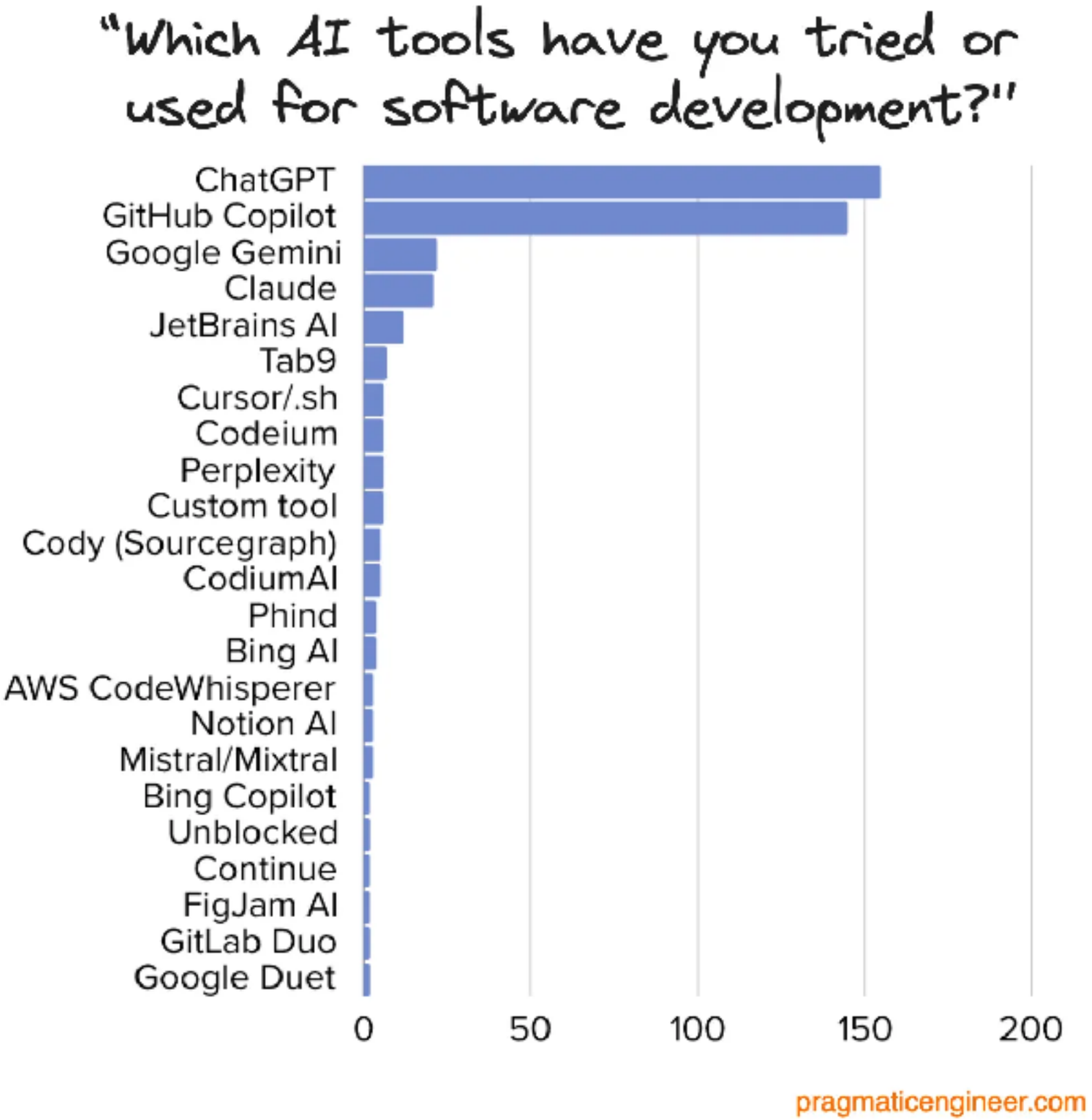
"Which AI tools have you tried or used for software development?"



https://newsletter.pragmaticengineer.com/p/ai-tooling-2024



https://newsletter.pragmaticengineer.com/p/ai-tooling-2024

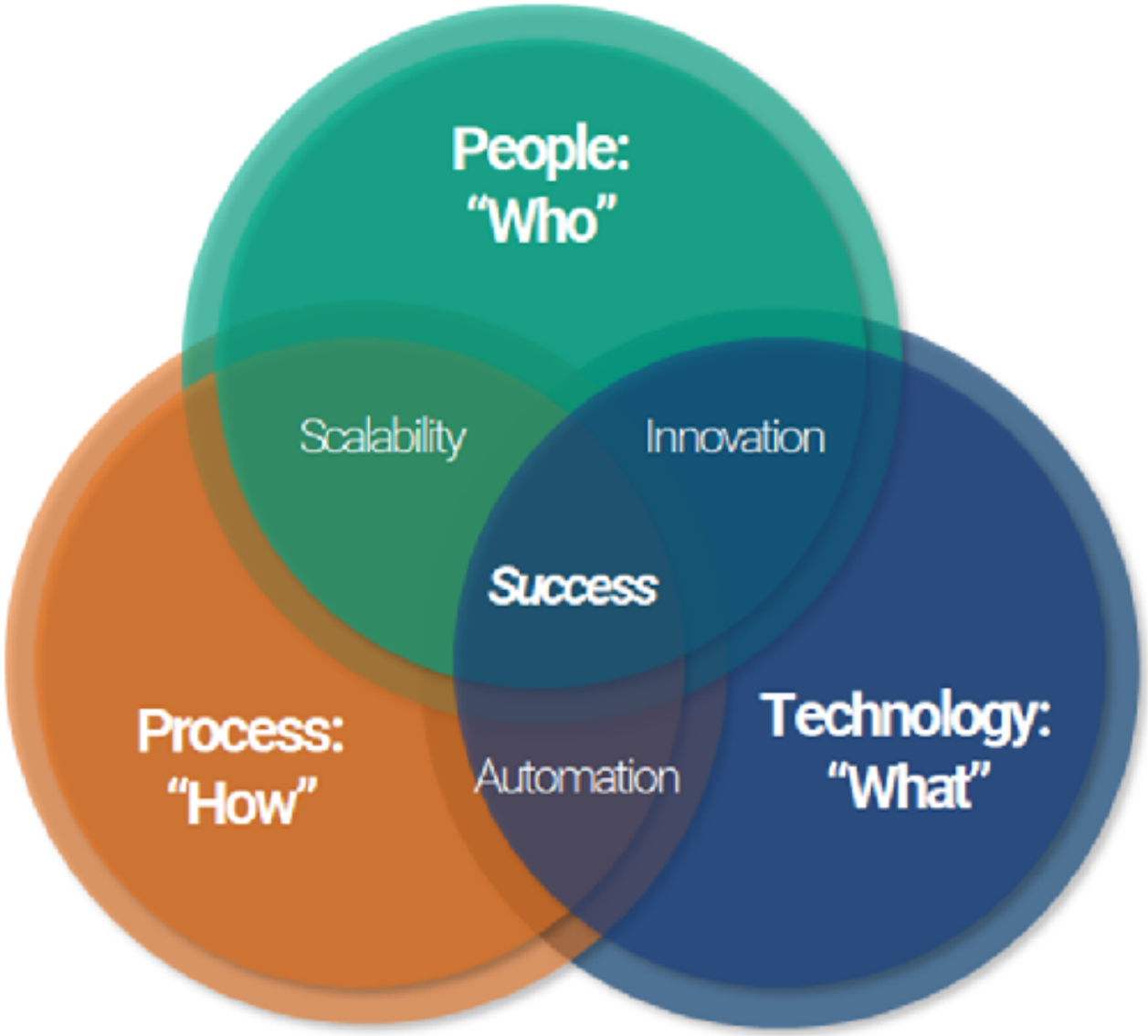


https://newsletter.pragmaticengineer.com/p/ai-tooling-2024

Generative AI isn't just a tool

it's your **team member**

3 Pillar of Software Development

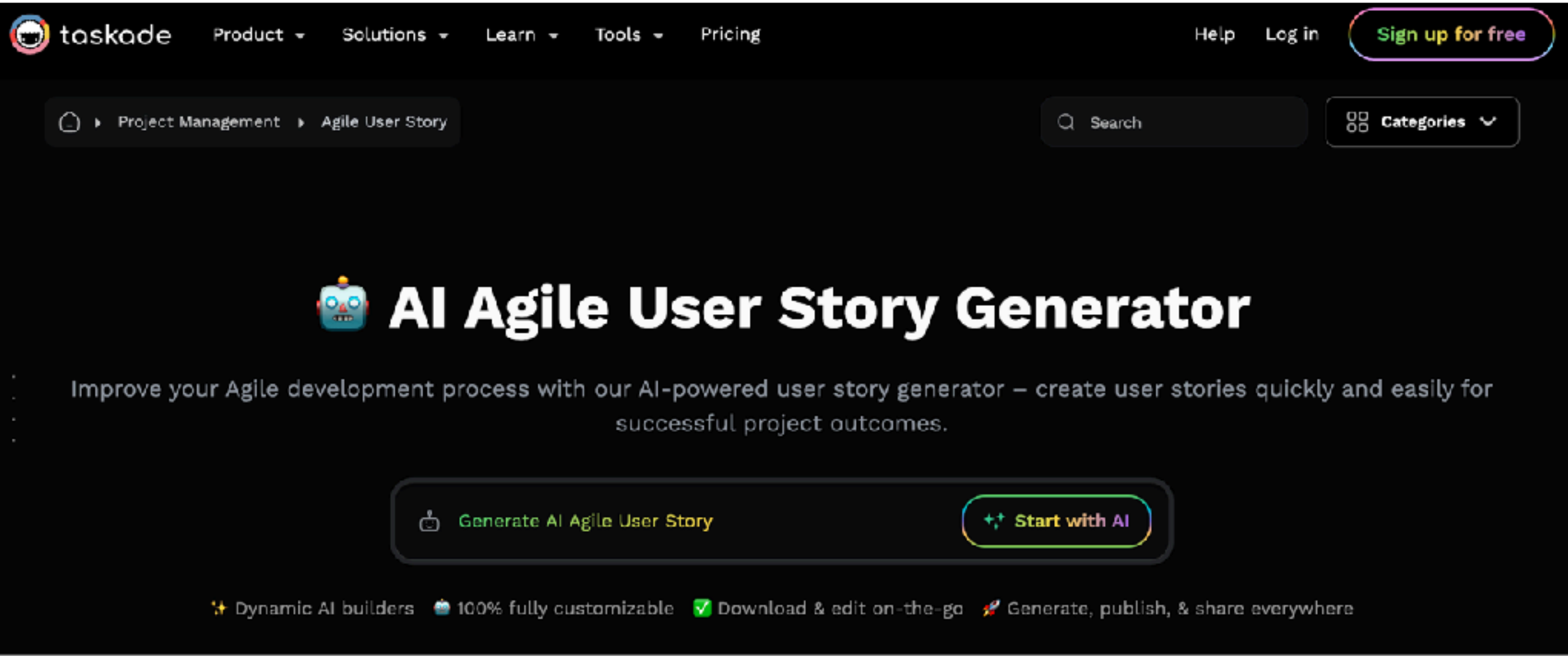


Requirement and Analysis

Requirement and Analysis

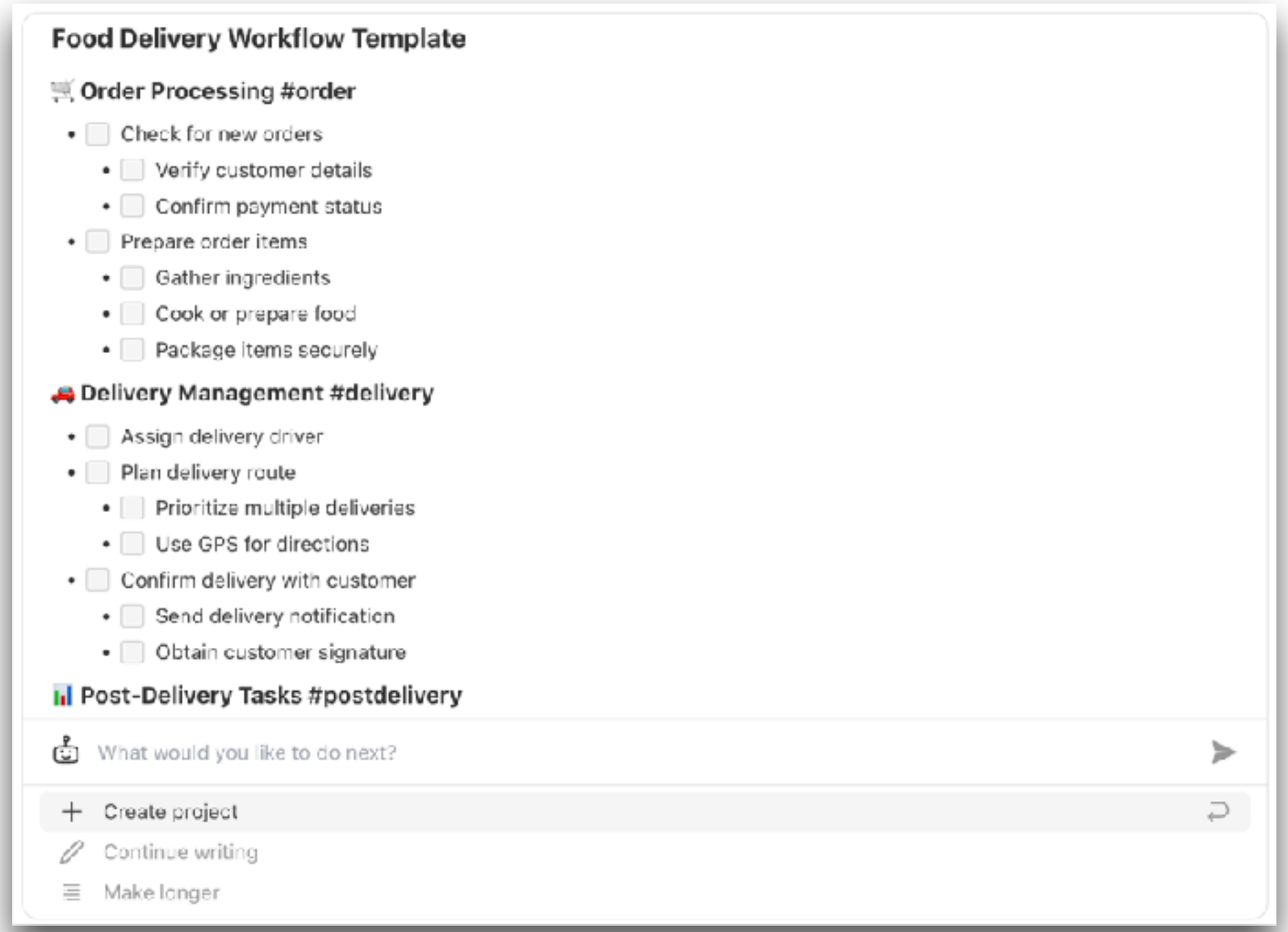
Requirements writing and analysis
User story generation

Agents AI for Automated tasks



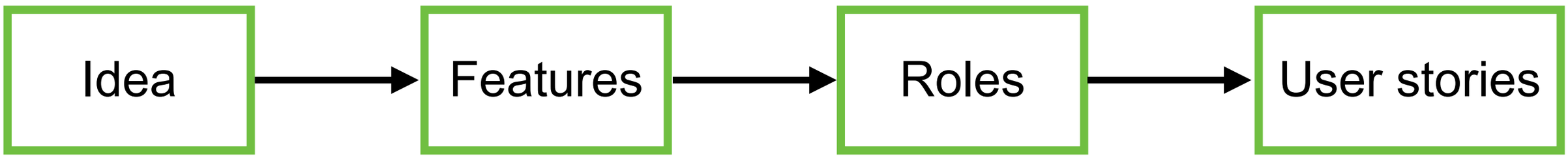
https://www.taskade.com/generate/project-management/agile-user-story

Example with food delivery



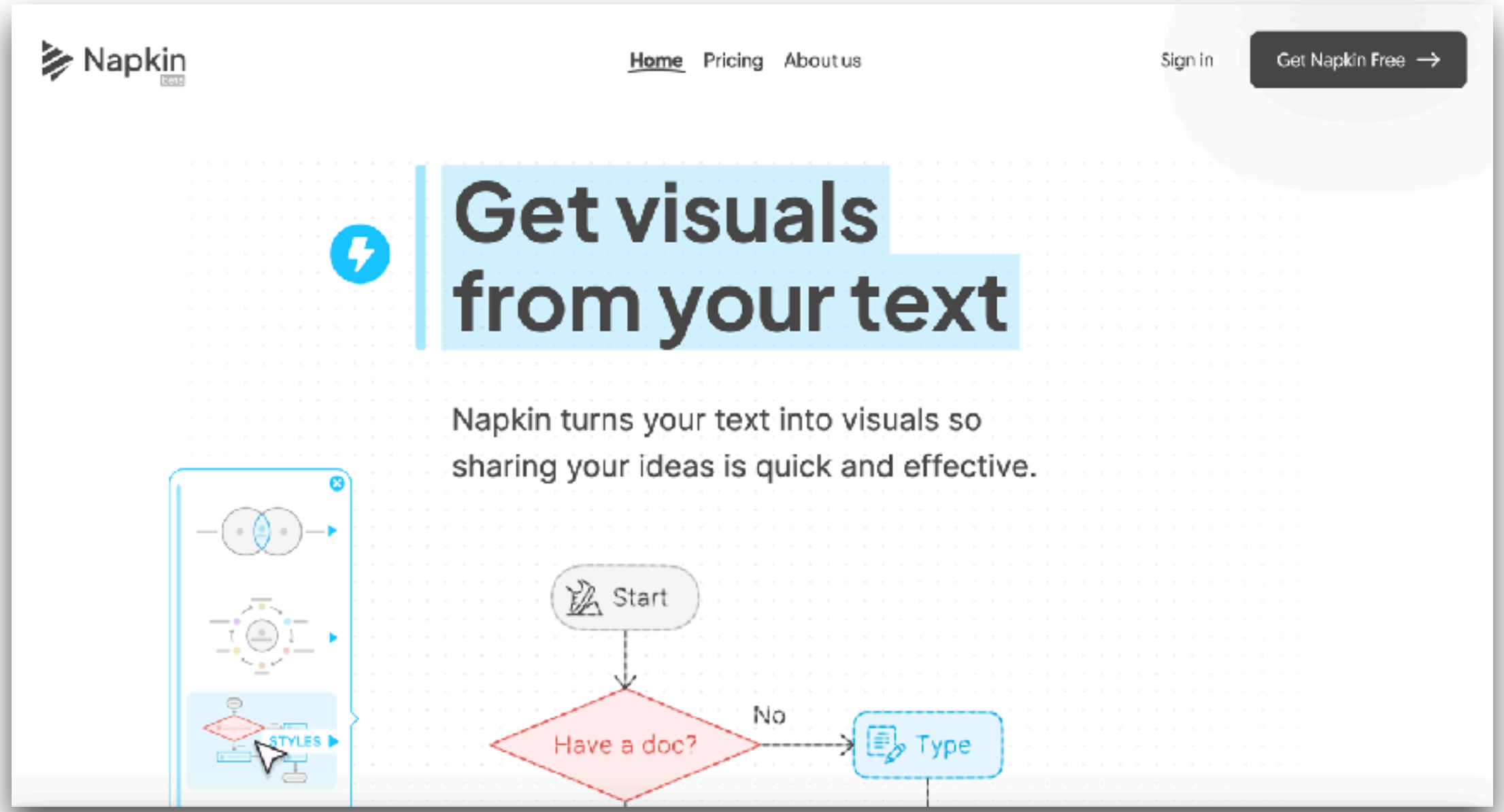
https://www.taskade.com/generate/project-management/agile-user-story

User Story Generator



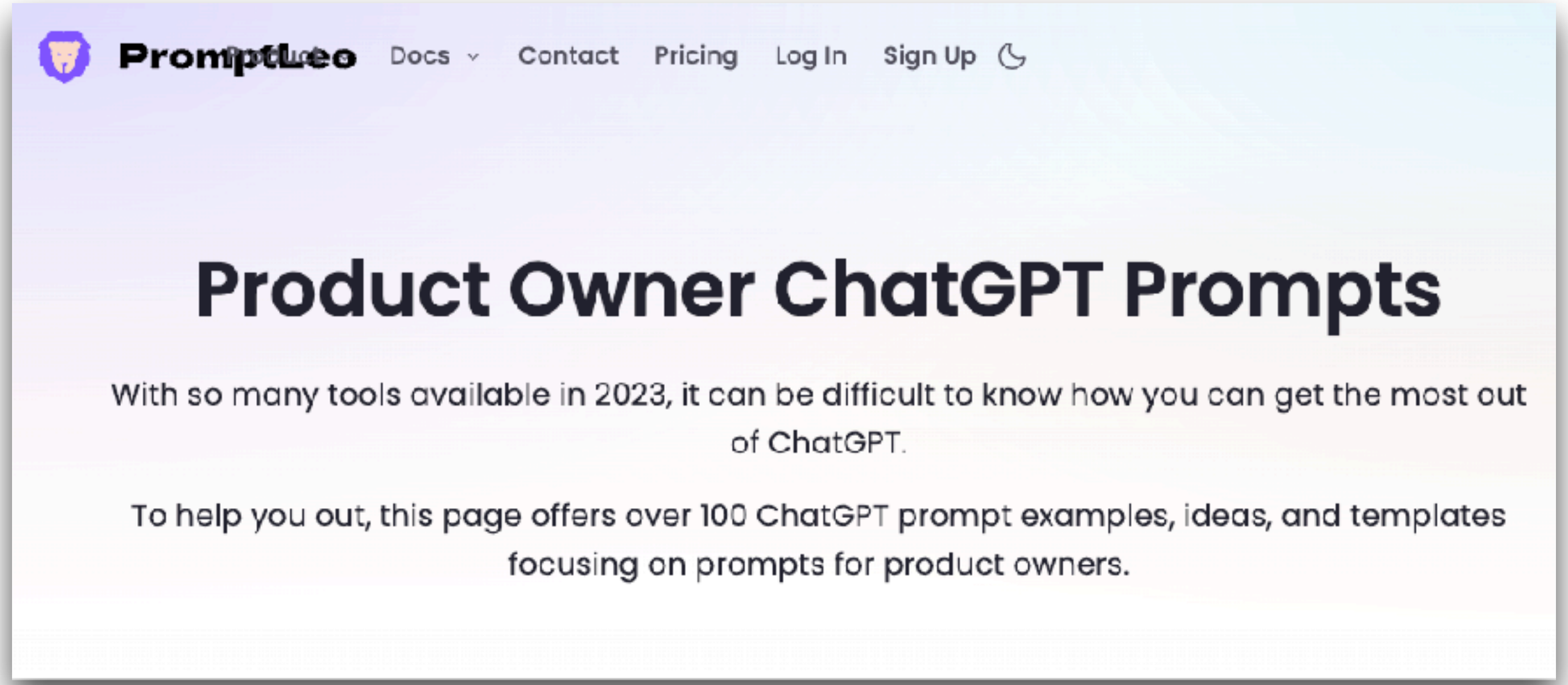
https://userstorygenerator.ai/

Napkin



https://www.napkin.ai/

Product Owner ChatGPT Prompt



https://promptleo.com/prompt/chatgpt/product-owner

Requirement analysis

Clarify of User requirement ?

https://github.com/up1/workshop-ai-with-technical-team/wiki/Requirement-analysis

Design Process

Design

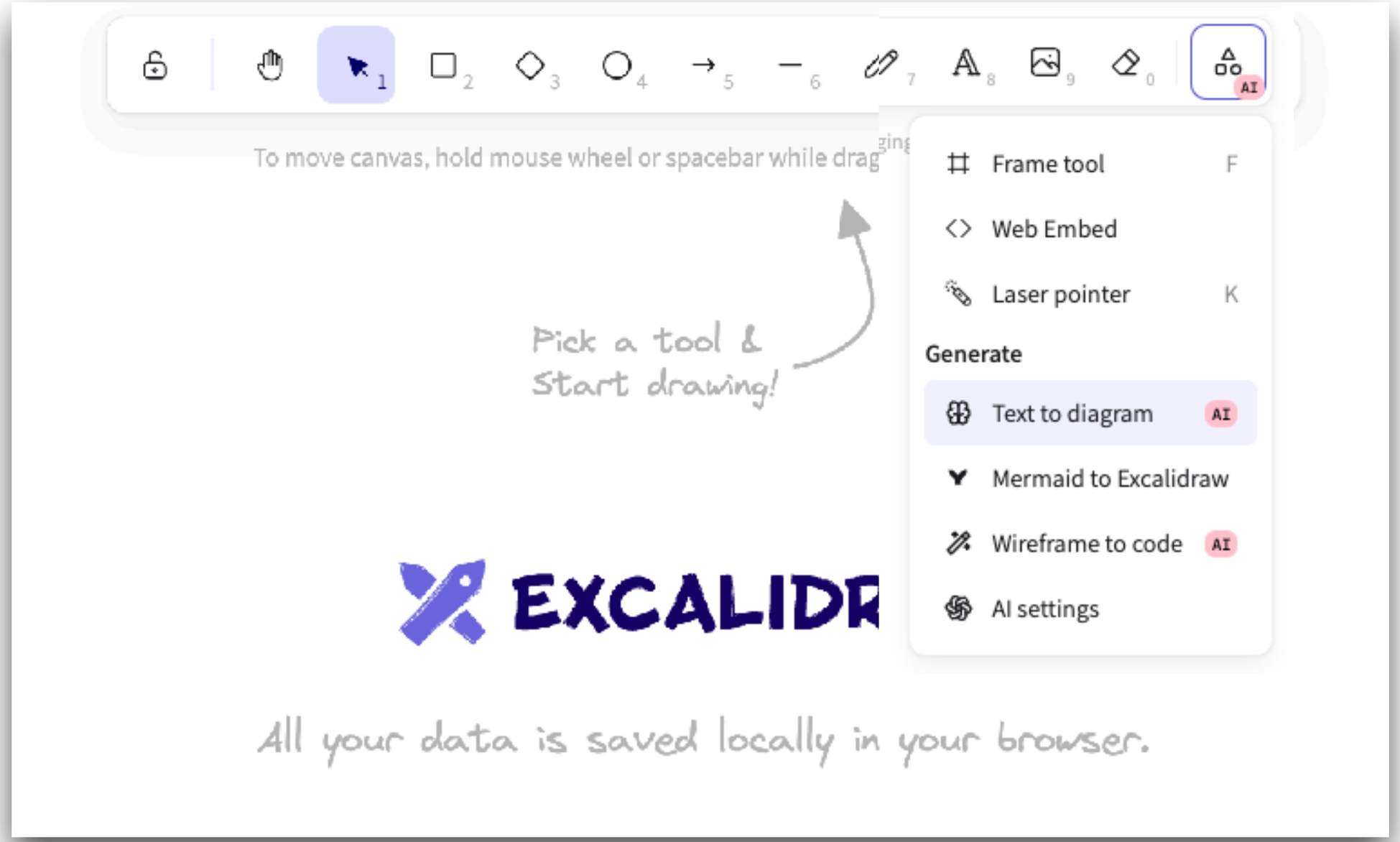
Architecture writing assistance

Sequence flow diagram generation

Data modeling

UX/UI design assistance

Excalidraw with AI



https://excalidraw.com/

Demo

Text to diagram AI Beta | Mermaid

Currently we use Mermaid as a middle step, so you'll get best results if you describe a diagram, workflow, flow chart, and similar.

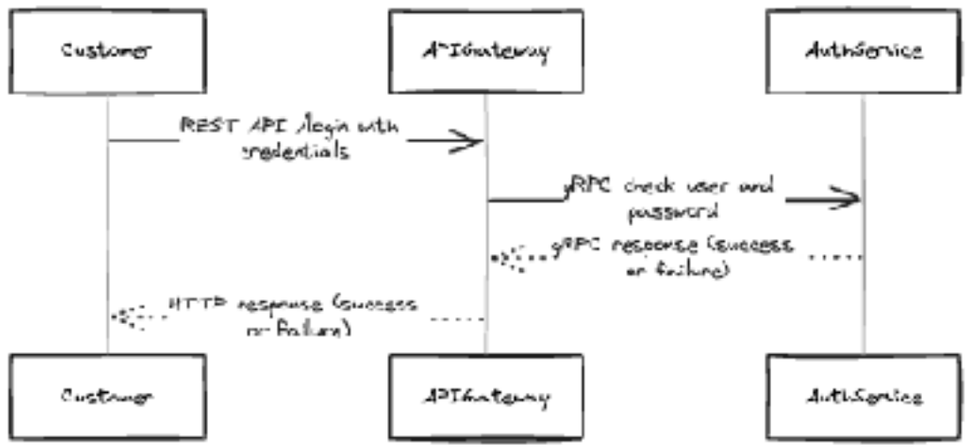
Prompt 9 requests left today

```
Try to generate authentication service from below
1. Customer call api gateway with REST API /login
2. Api gateway check user and password from auth service via
gRPC
3. API gateway send response to client
```

Generate → Cmd Enter

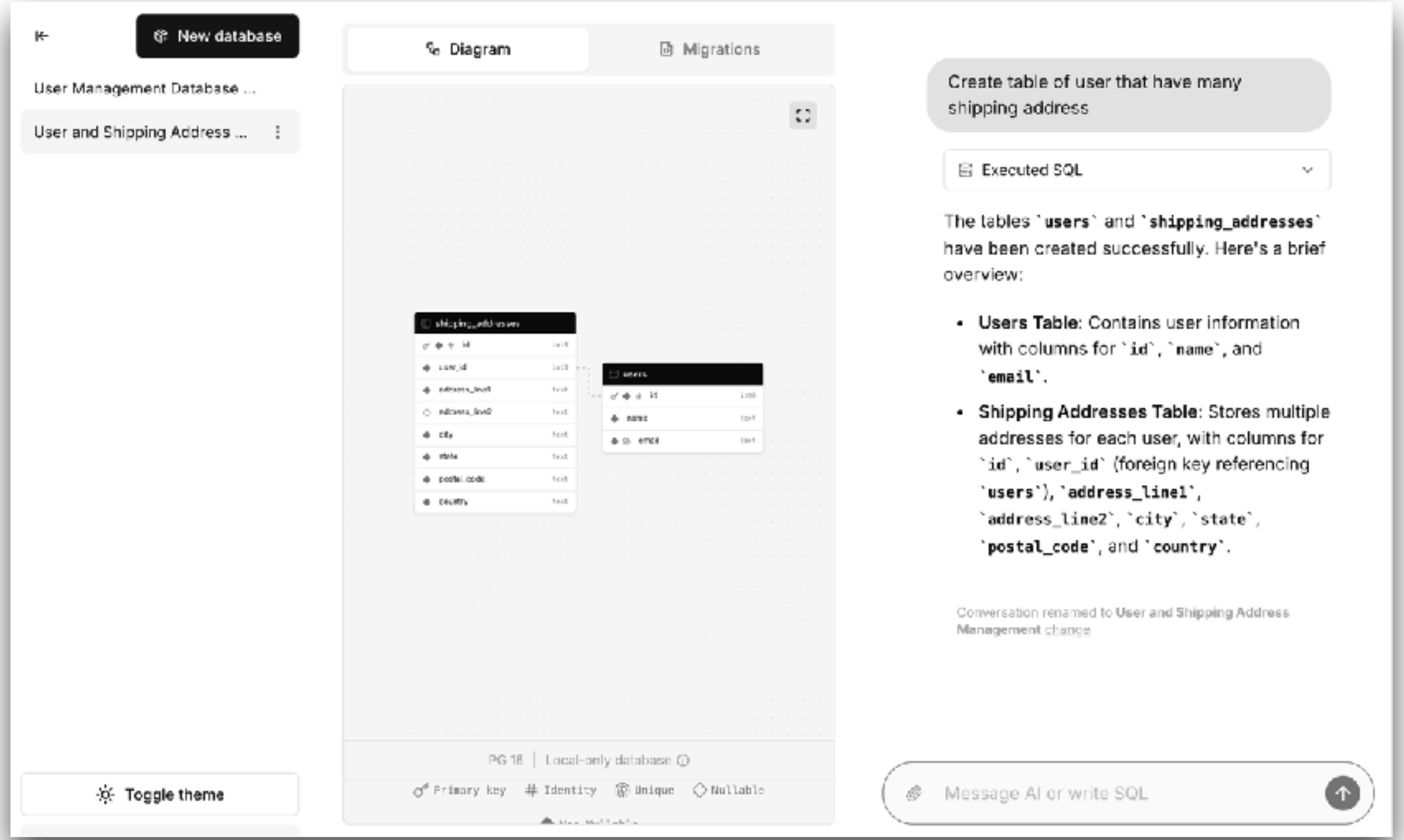
View as Mermaid →

Preview



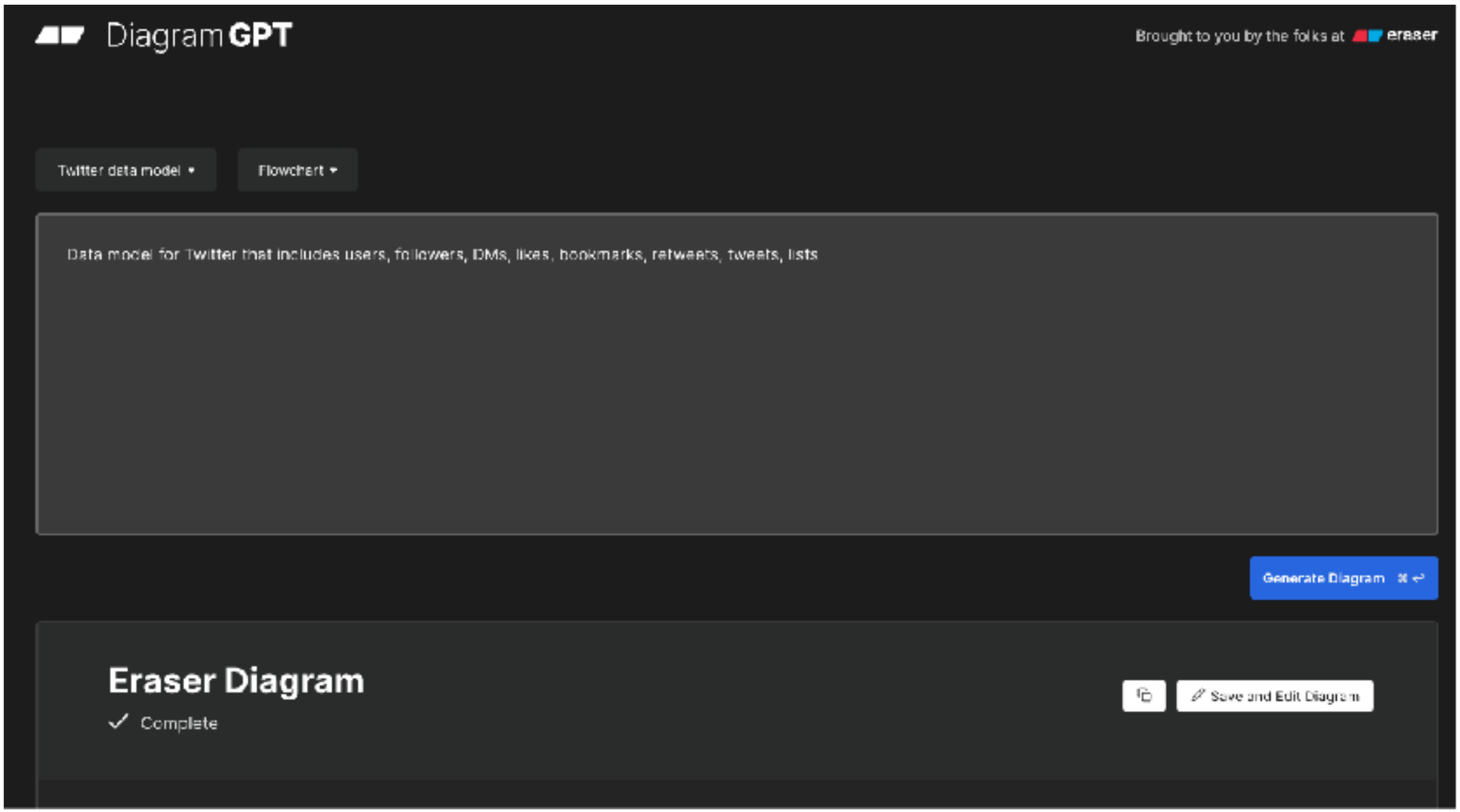
Insert →

Database



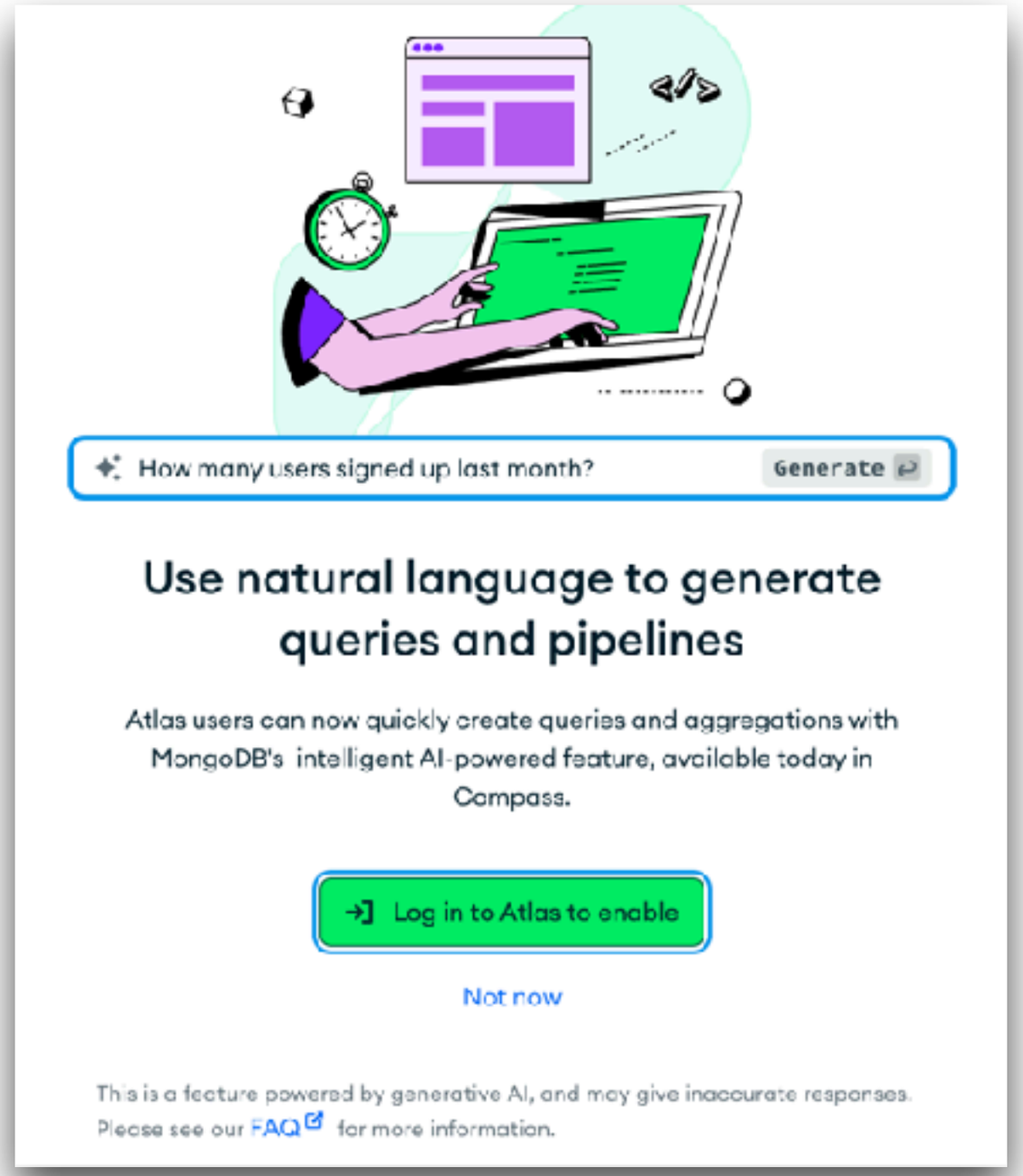
https://database.build/

DiagramGPT



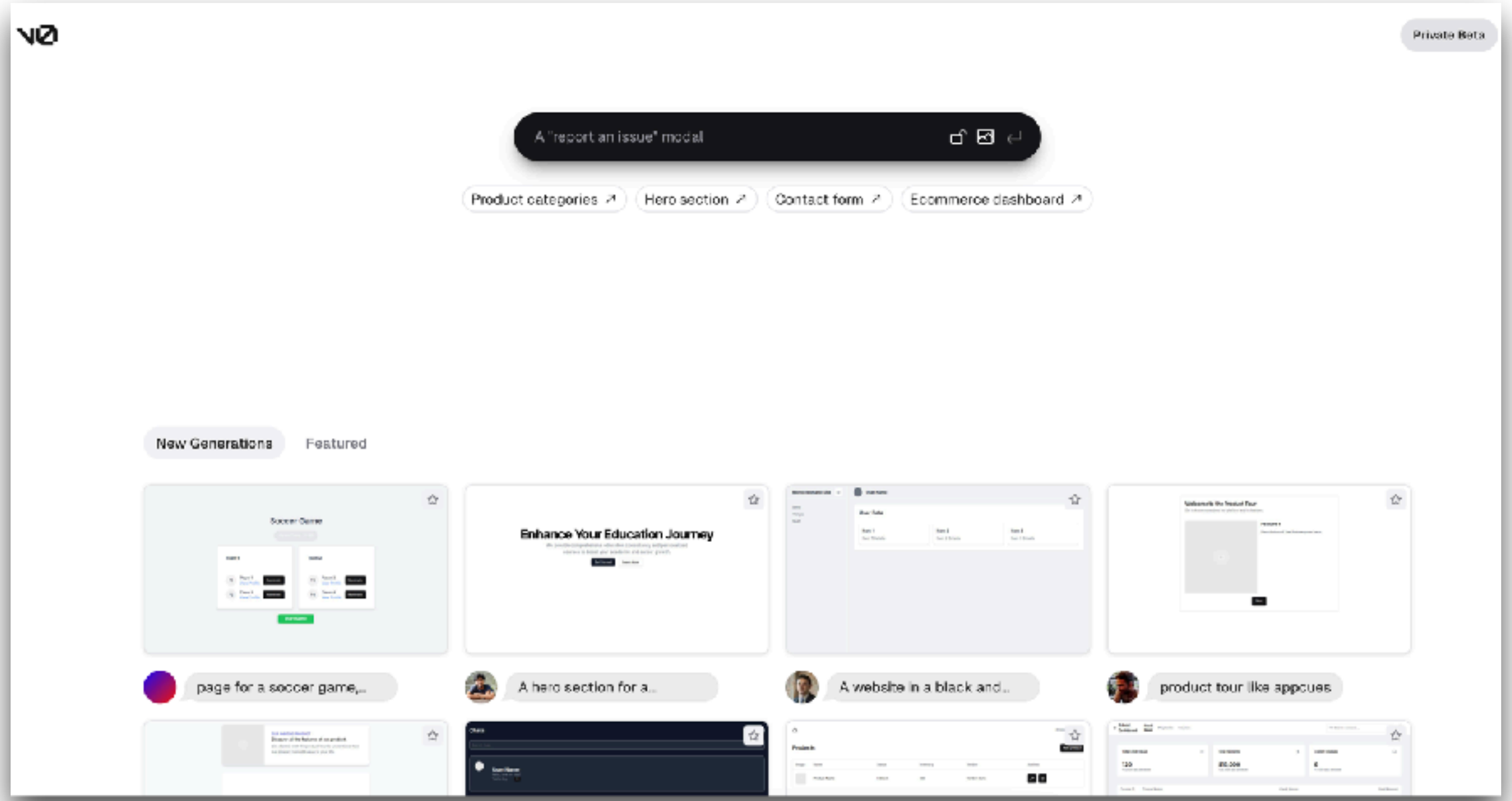
https://www.eraser.io/diagramgpt

MongoDB Compass



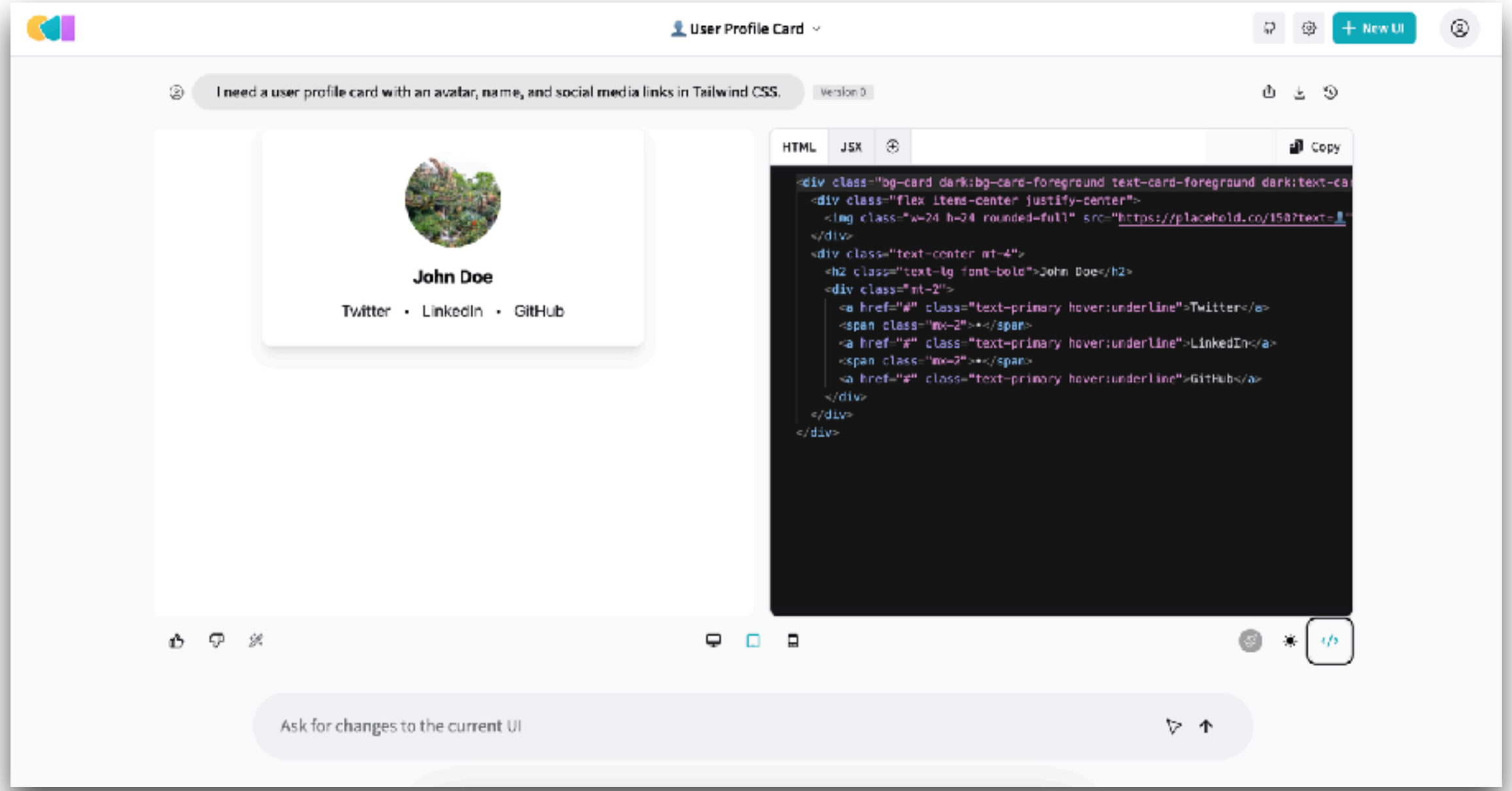
https://www.mongodb.com/products/tools/compass

V0.dev



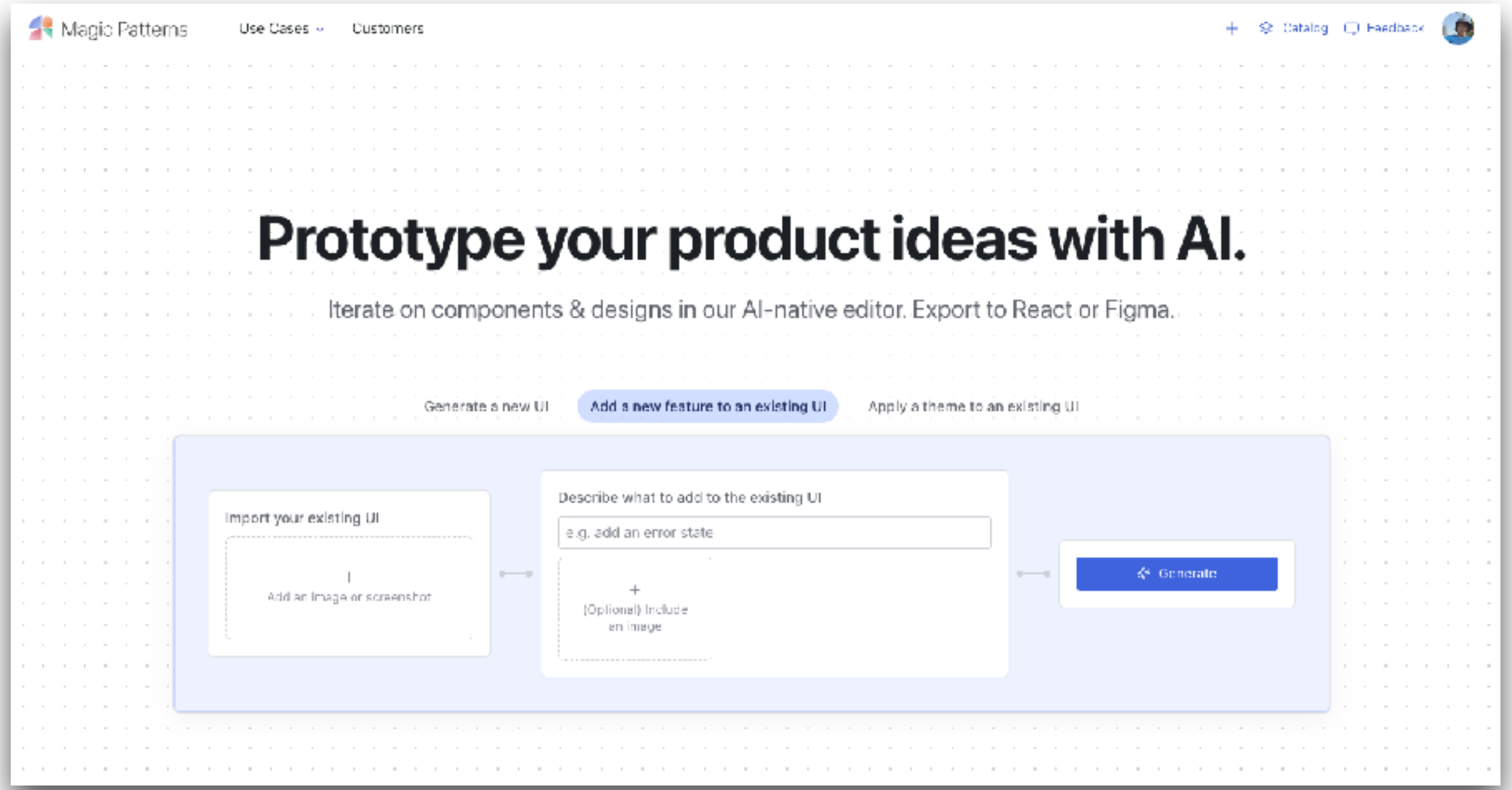
https://v0.dev/

OpenUI



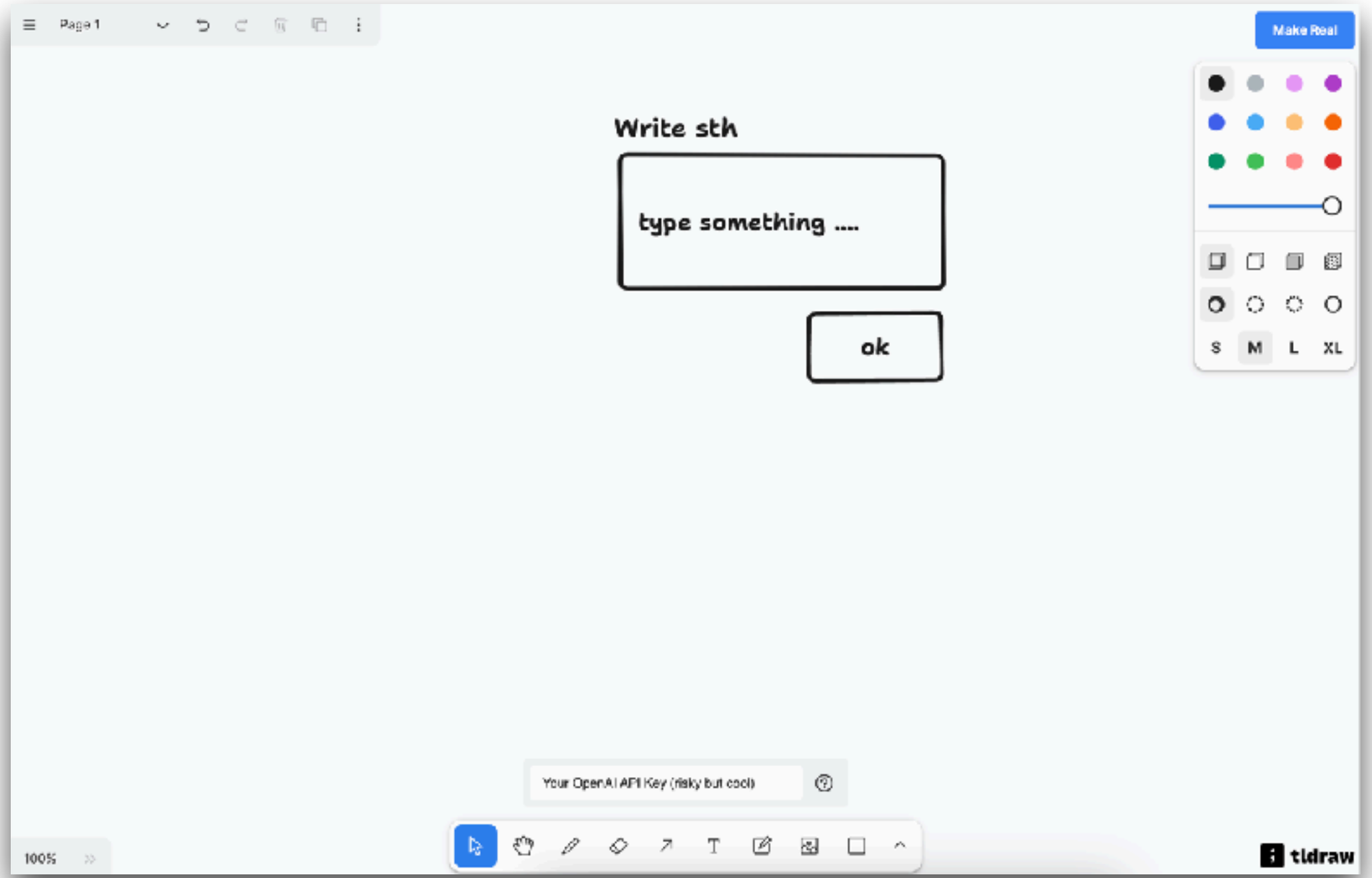
https://openui.fly.dev/

Magic Pattern



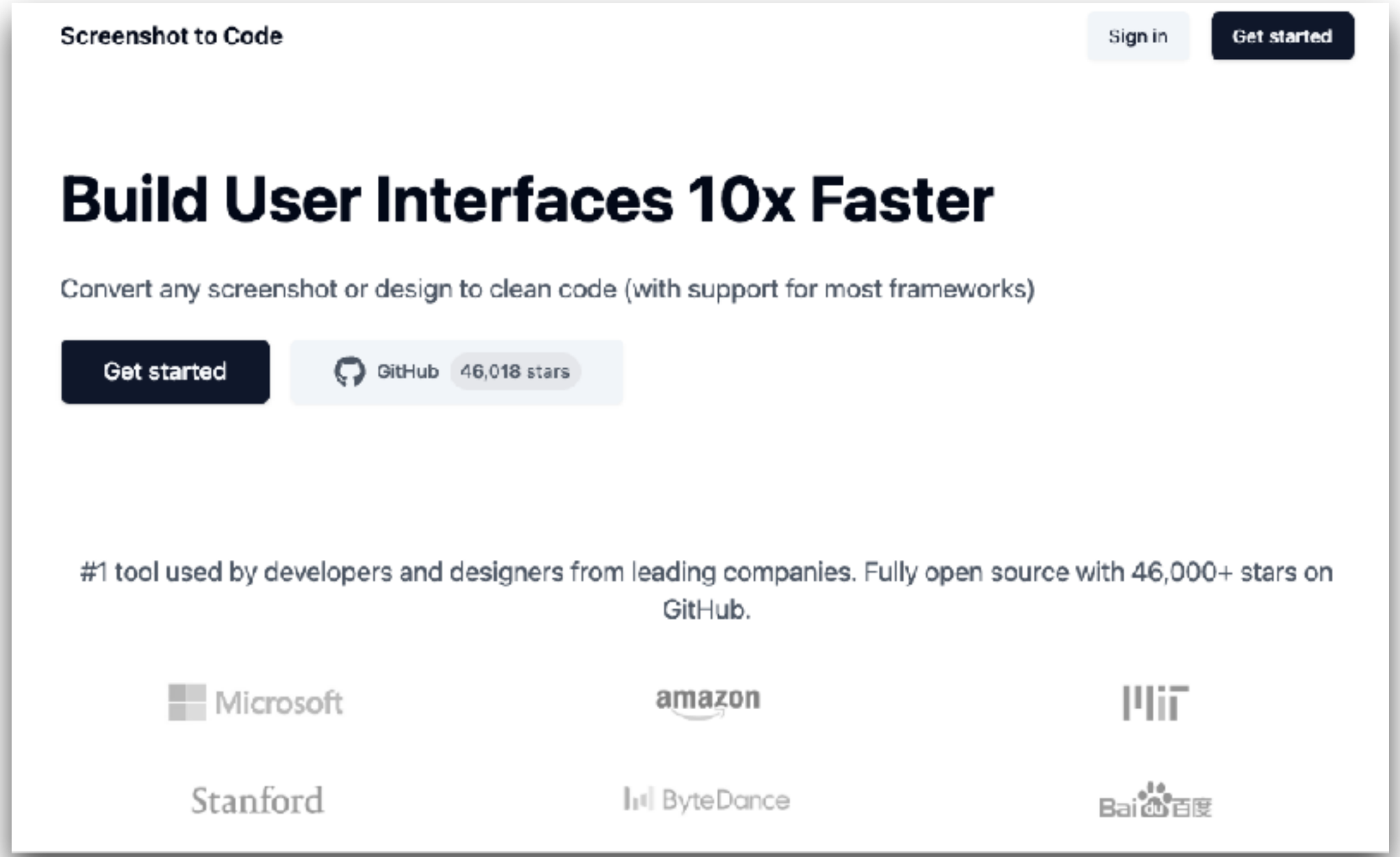
https://www.magicpatterns.com/

Make Real



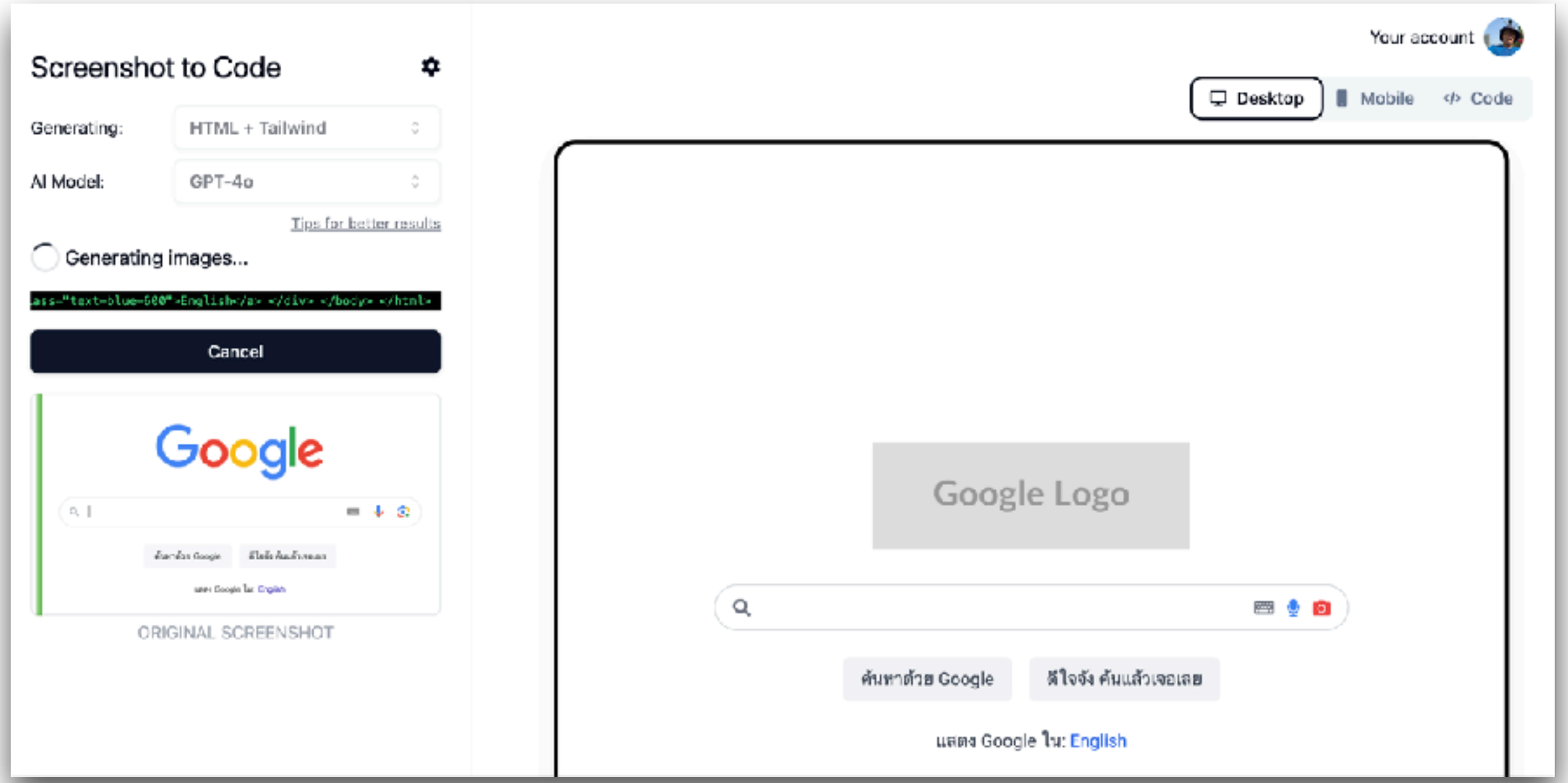
https://github.com/tldraw/make-real

Screenshot to Code



https://github.com/abi/screenshot-to-code

Demo



https://screenshottocode.com/

Development Process

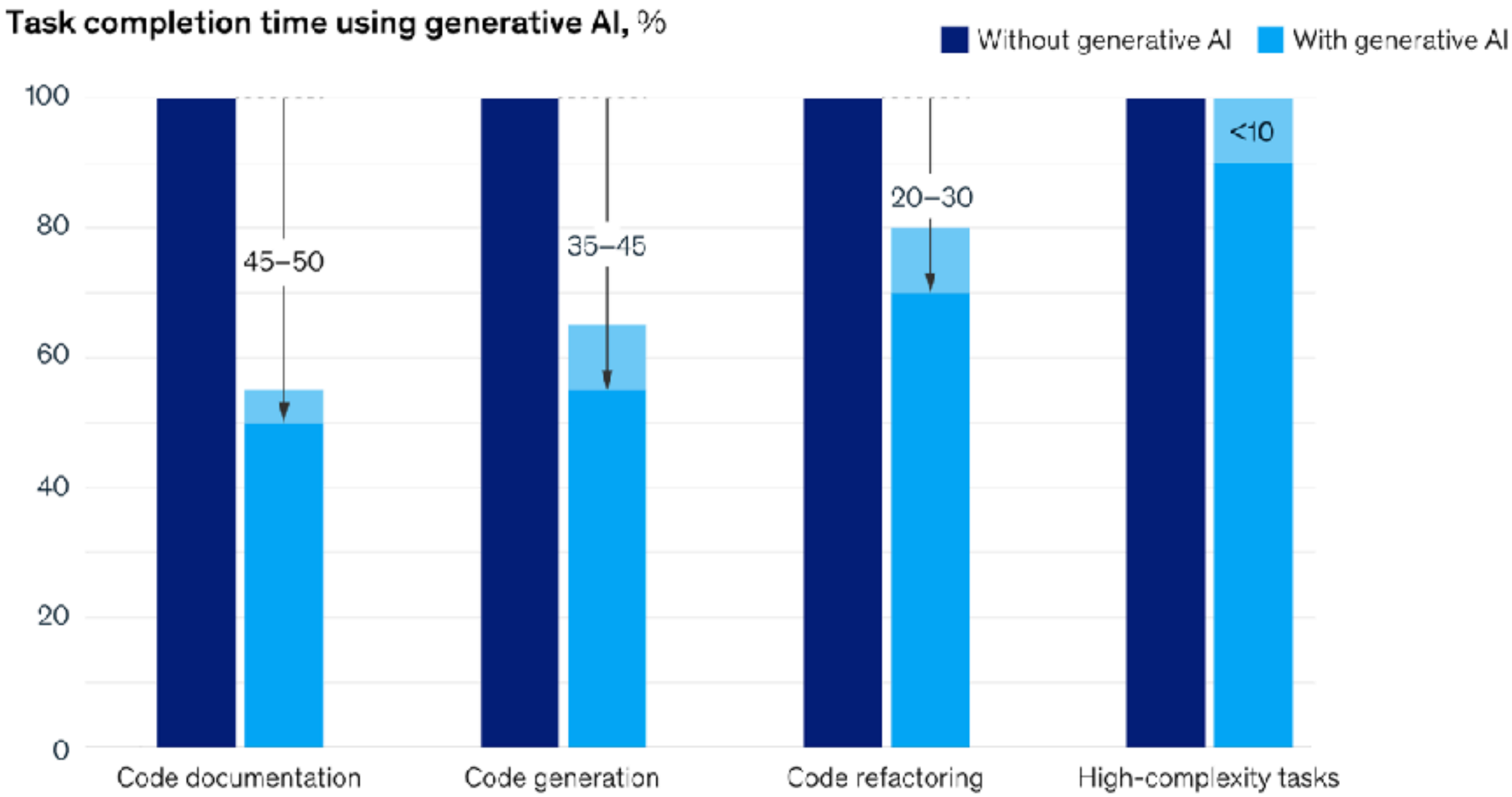
Develop

Requirement | Design | **Develop** | Testing | Deploy

Code generation
Review and explain code
Debugging code
Improve consistency
Code translation

Development

Generative AI can increase developer speed, but less so for complex tasks.



https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/unleashing-developer-productivity-with-generative-ai

Tools (Seamless with developer)

GitHub Copilot

Codium AI

aider

Continue

Bito

Category of Tools

Chat AI

Code AI

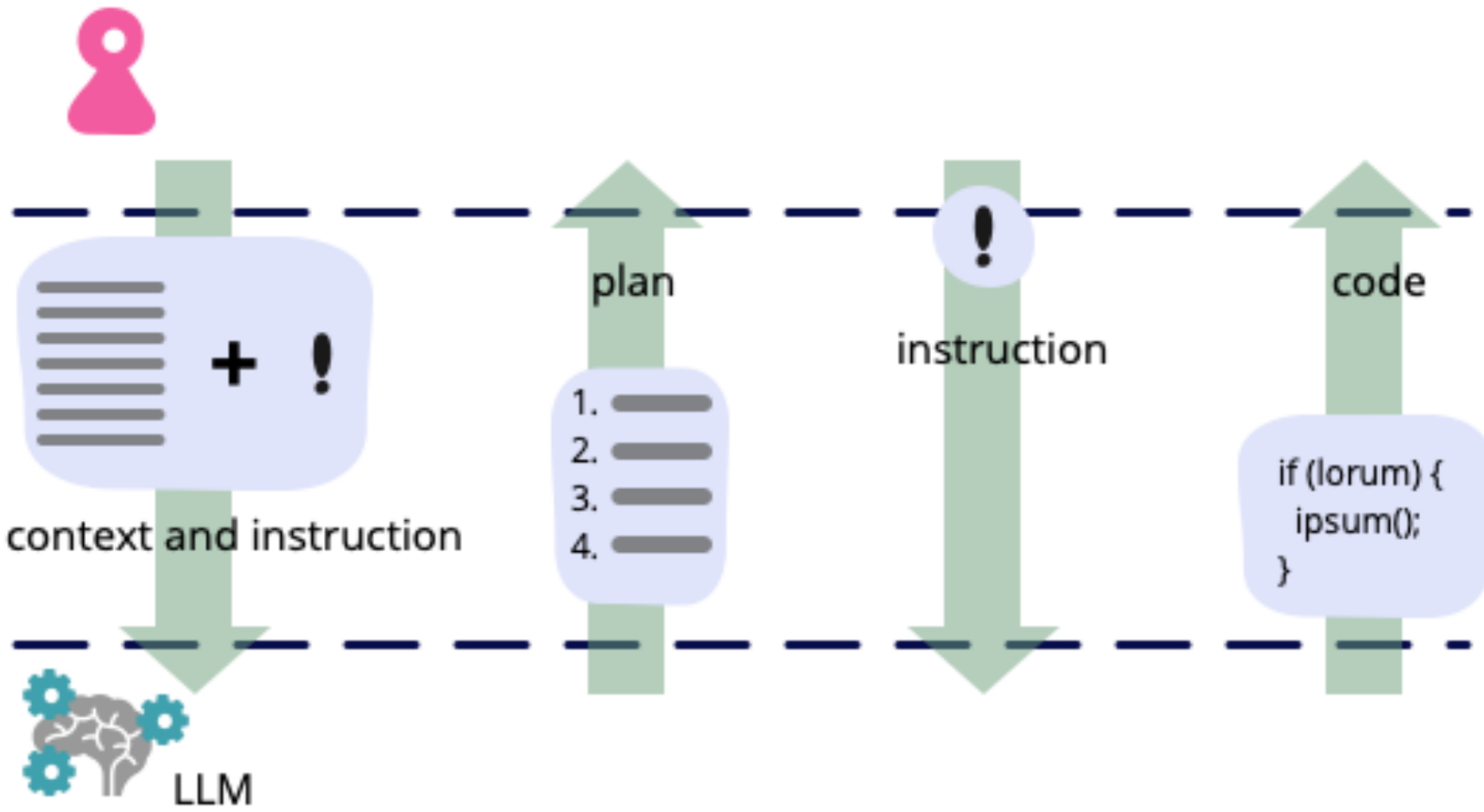
Agent AI

ChatGPT
Gemini
Claude.ai
Bing

Category of Tools

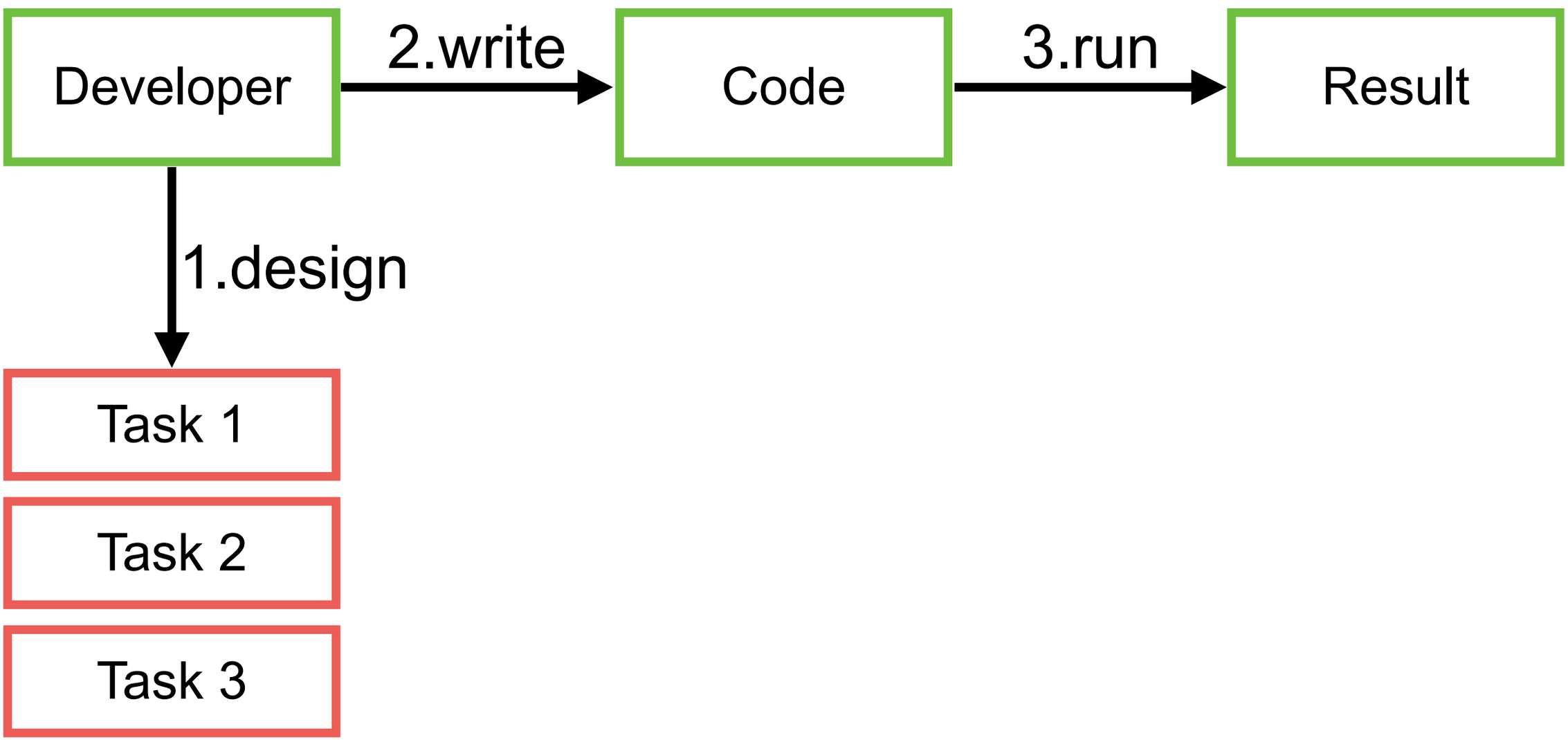
Chat AI	Code AI	AI Agent Public and Local
ChatGPT Gemini Claude.ai Bing	GitHub Copilot	Aider Continue Codium

Development

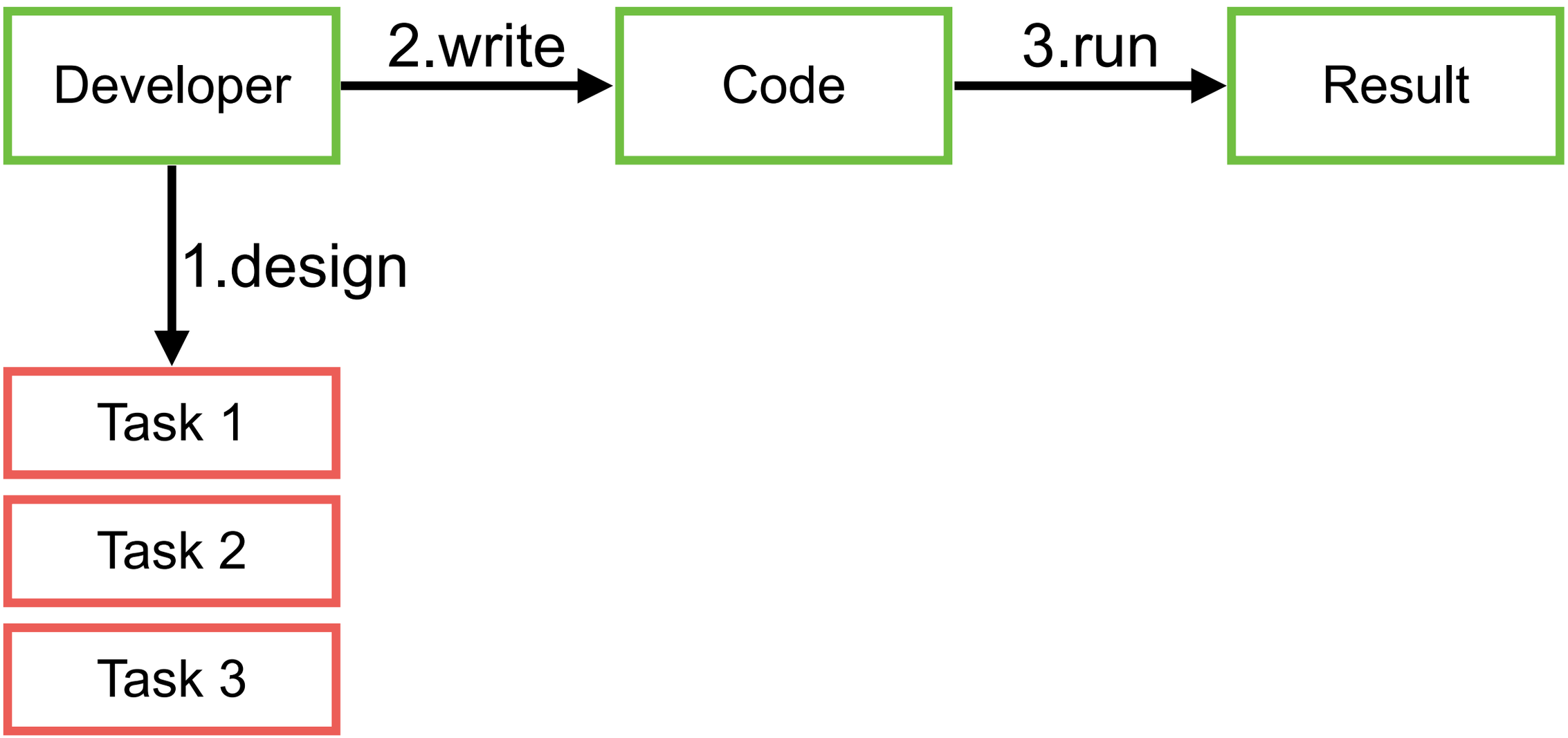


https://martinfowler.com/articles/2023-chatgpt-xu-hao.html

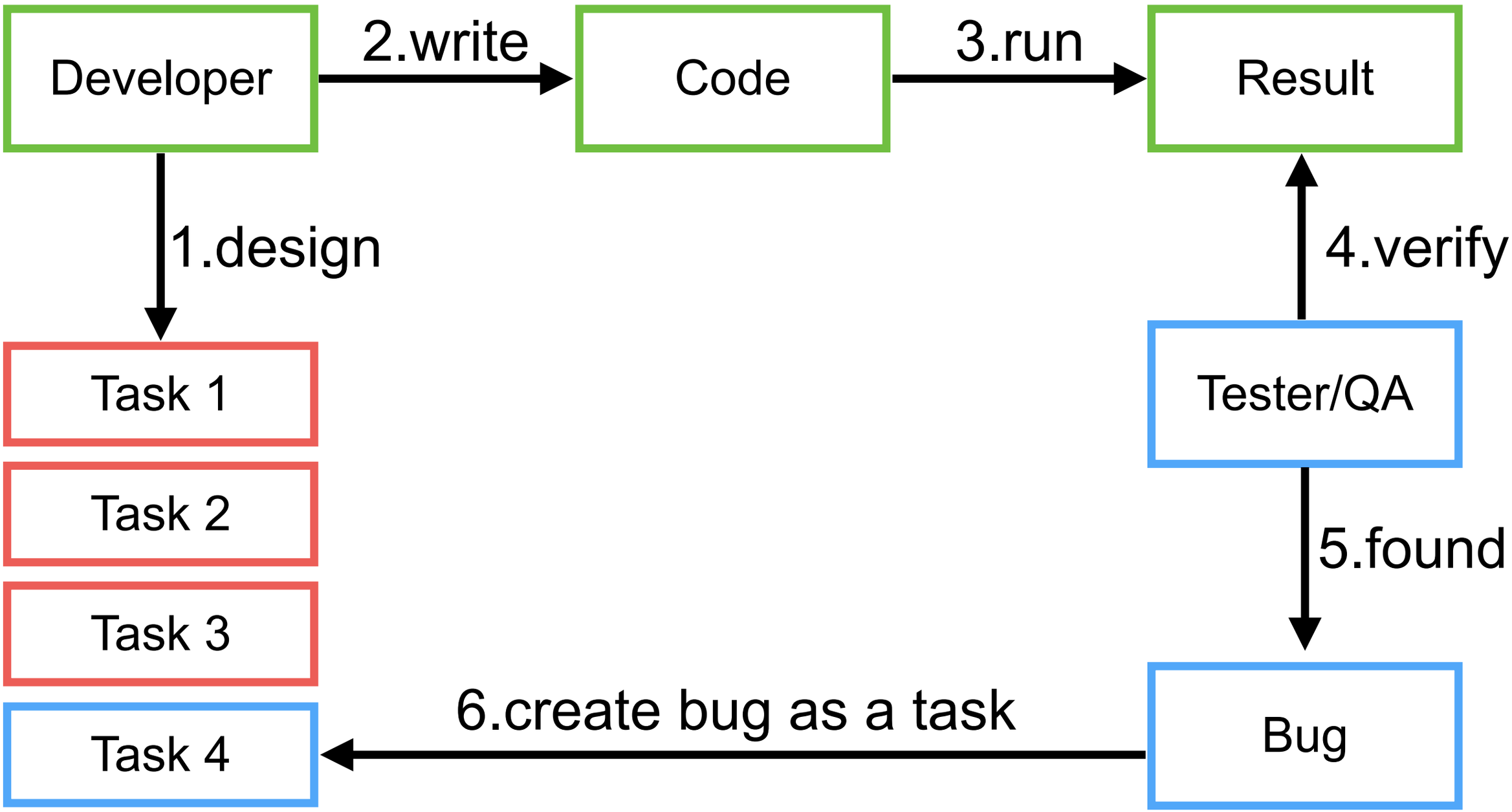
Development



Development



Development + Testing



Main Features

Ask question

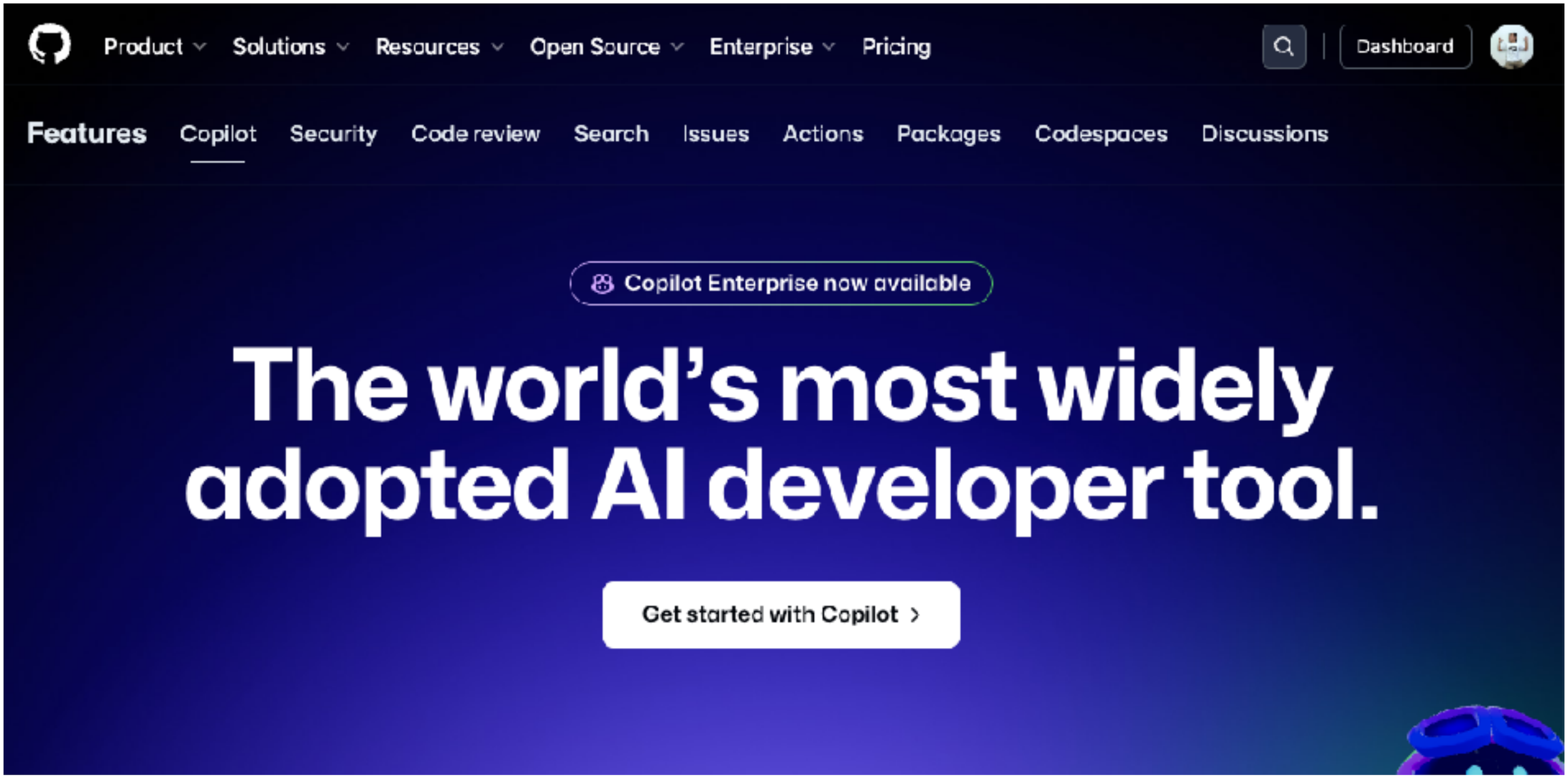
Generate code

Refactor code

Document code

Find problems in your code

GitHub Copilot



https://github.com/features/copilot

Using GitHub Copilot Chat correlates with better code quality

85% of developers felt more confident in their code quality when authoring code with GitHub Copilot and GitHub Copilot Chat

85%

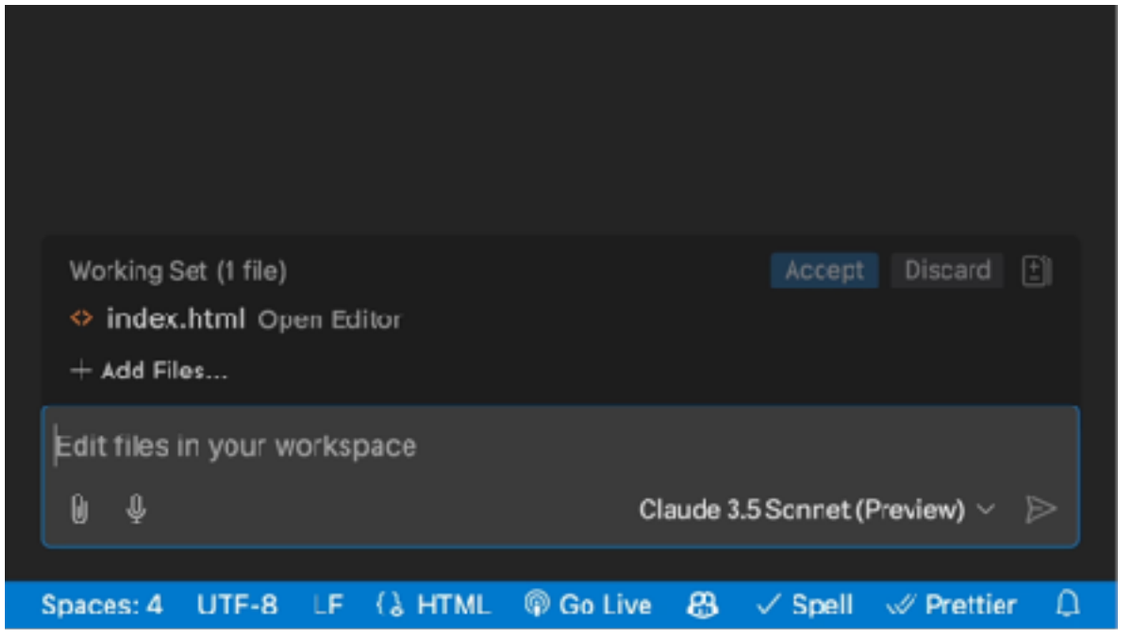
Code reviews were more actionable and completed 15% faster than without GitHub Copilot Chat

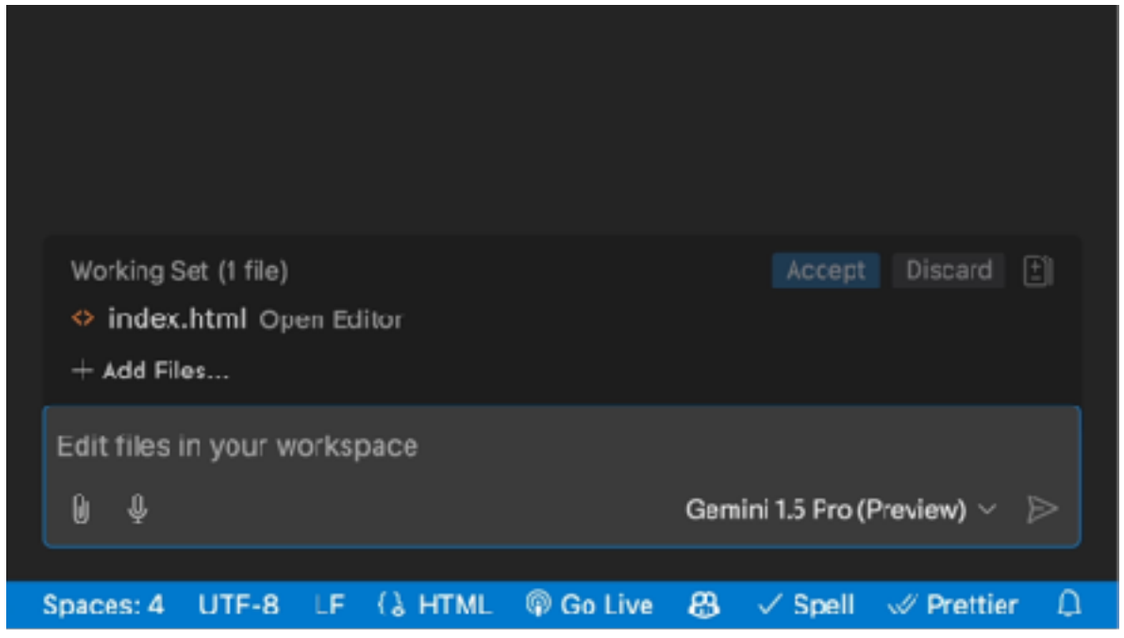
15%

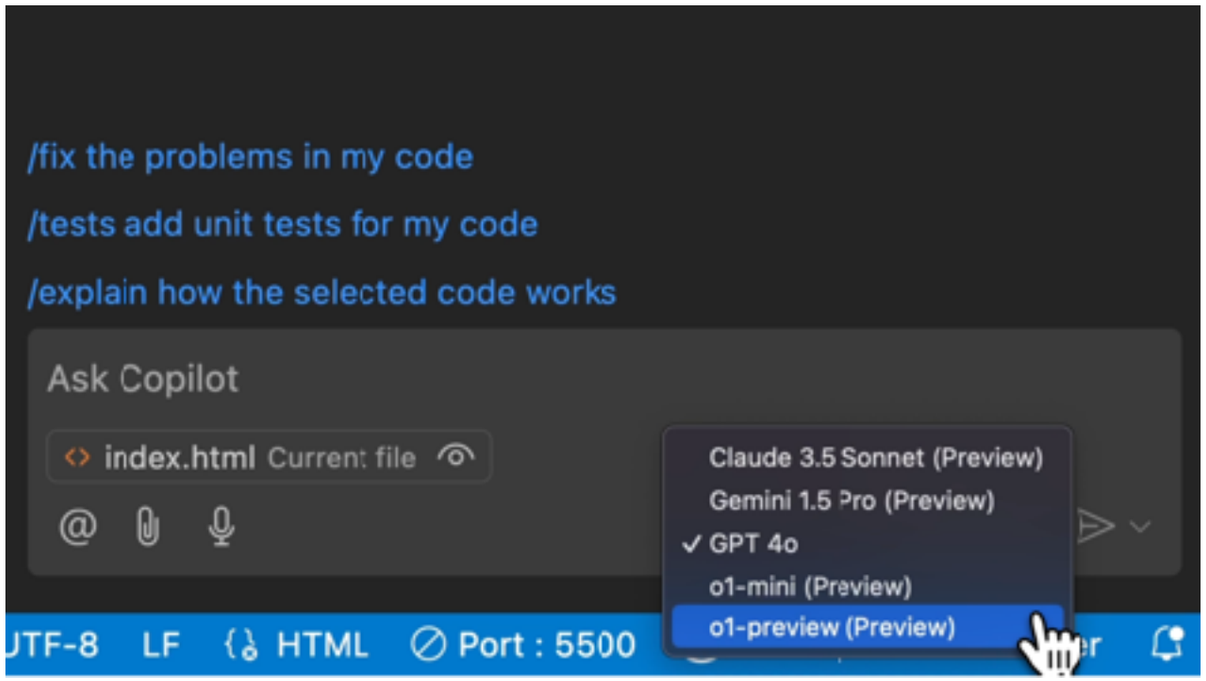
88% of developers reported maintaining flow state with GitHub Copilot Chat

88%

GitHub Copilot + Multi-models

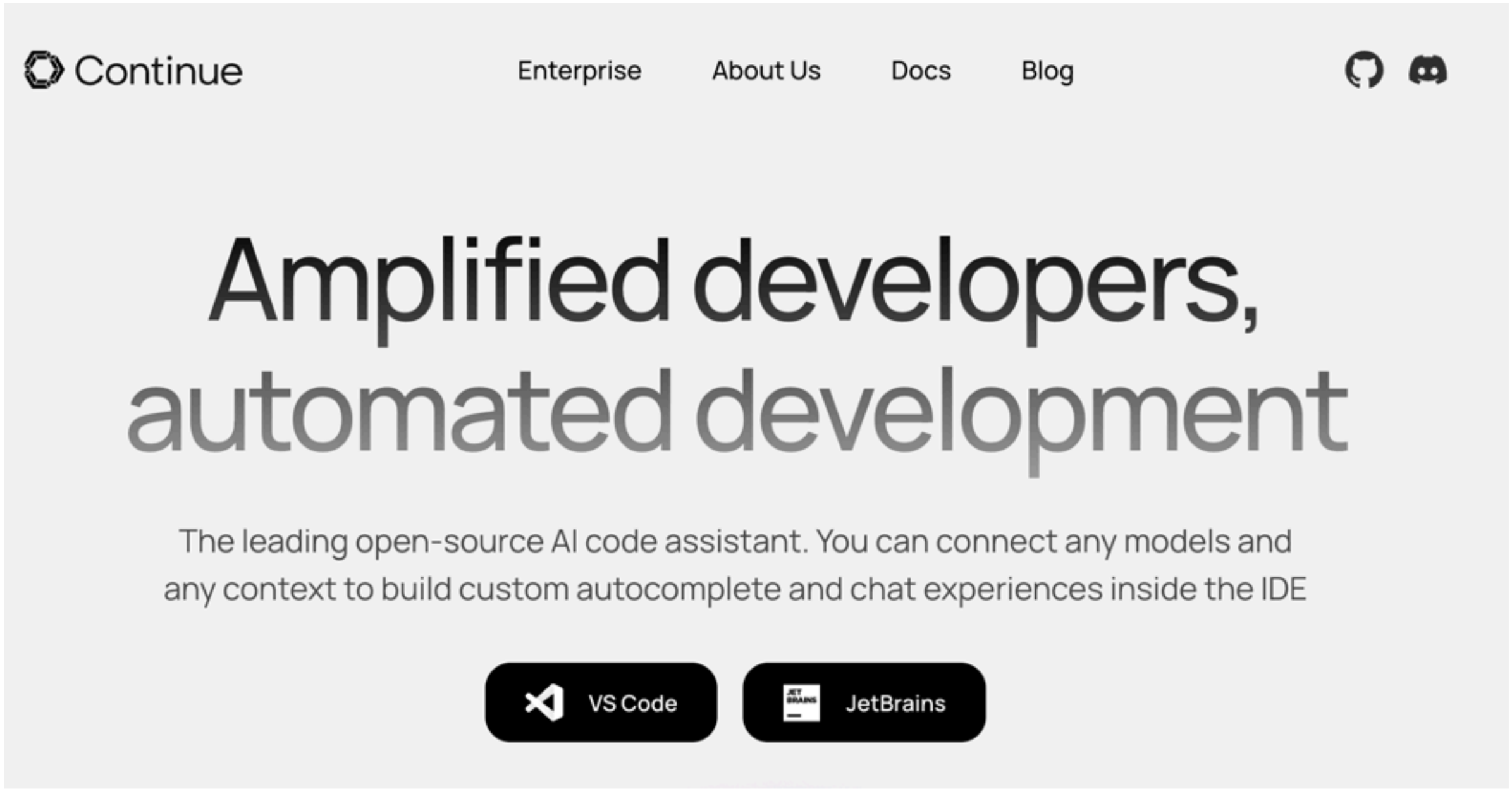






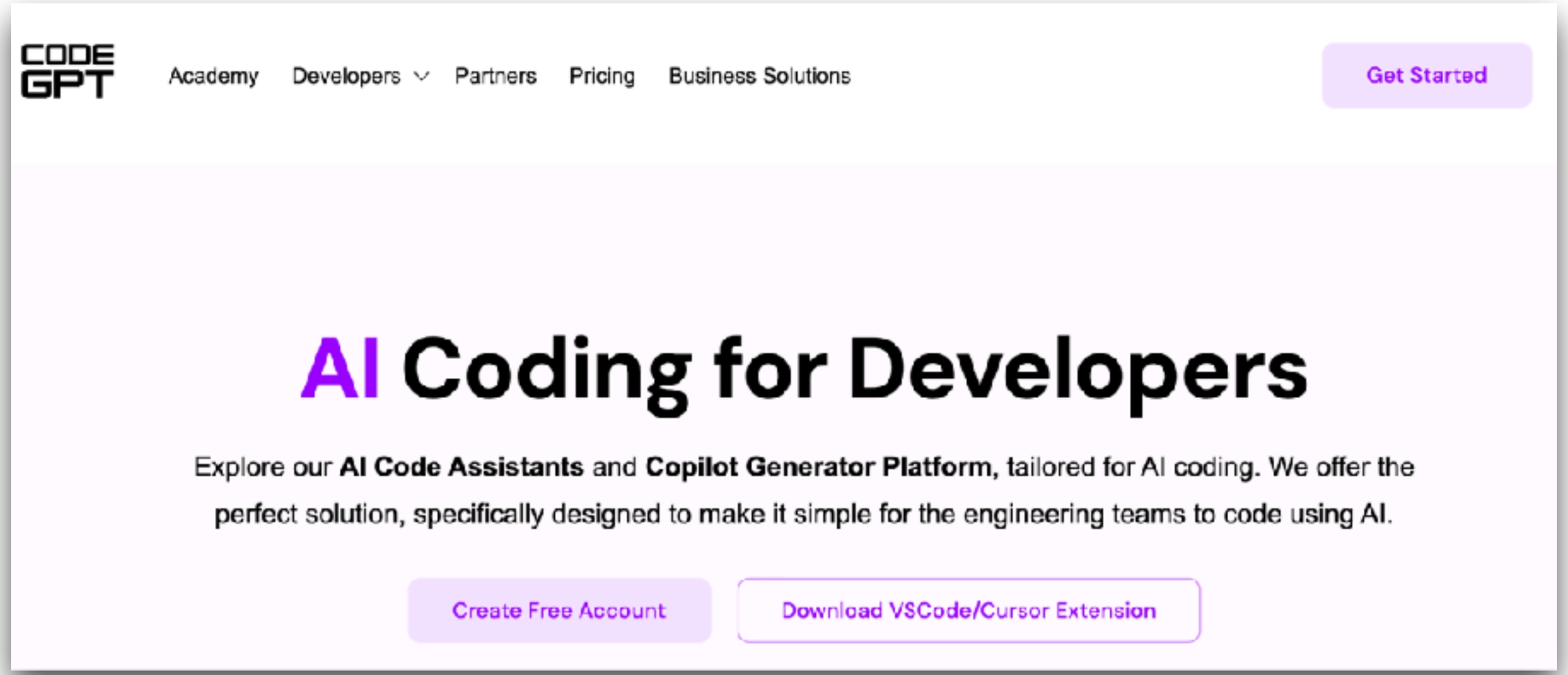
https://github.blog/news-insights/product-news/bringing-developer-choice-to-copilot/

Auto pilot with Code



https://github.com/continuedev/continue

CodeGPT

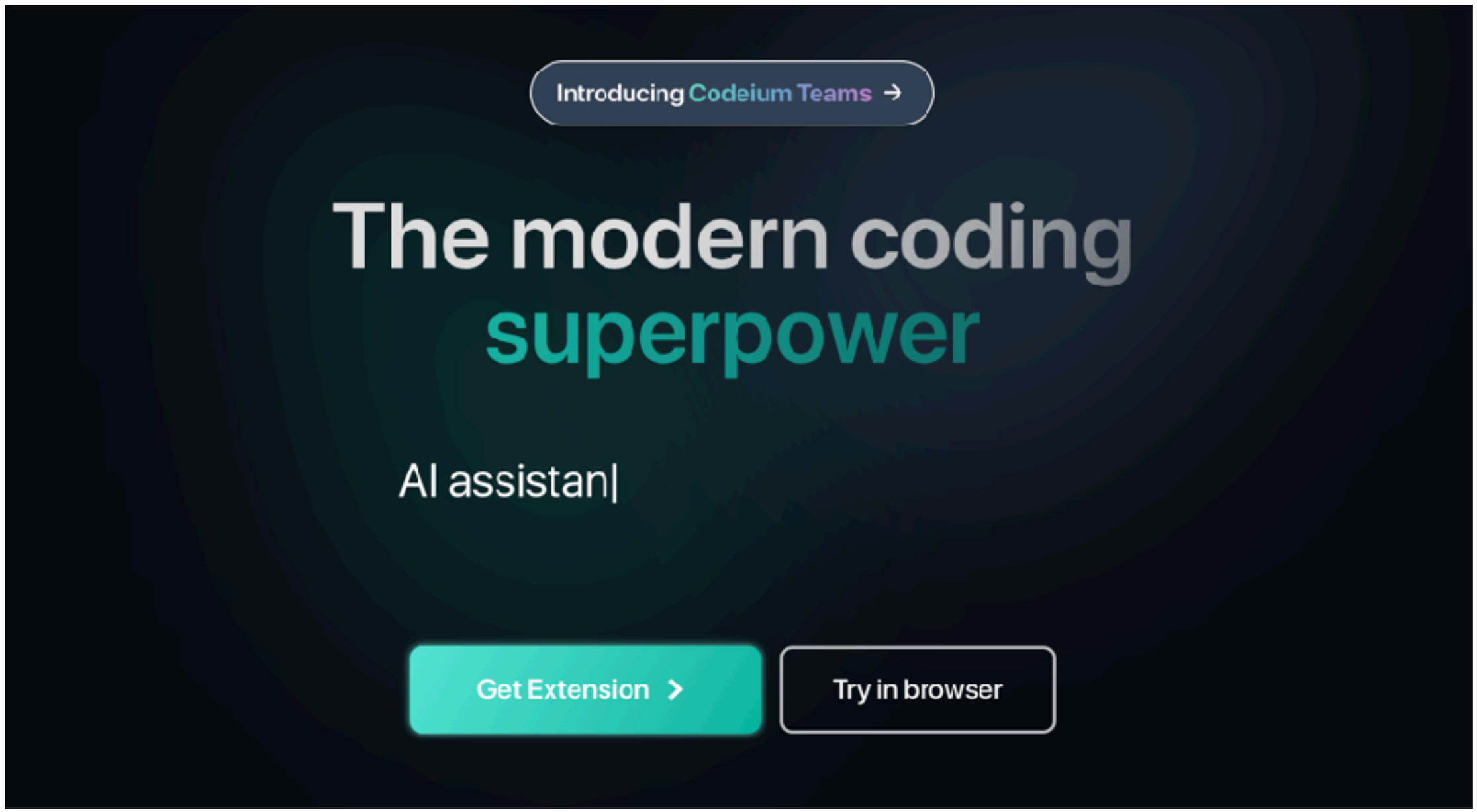


https://codegpt.co/

Code Review

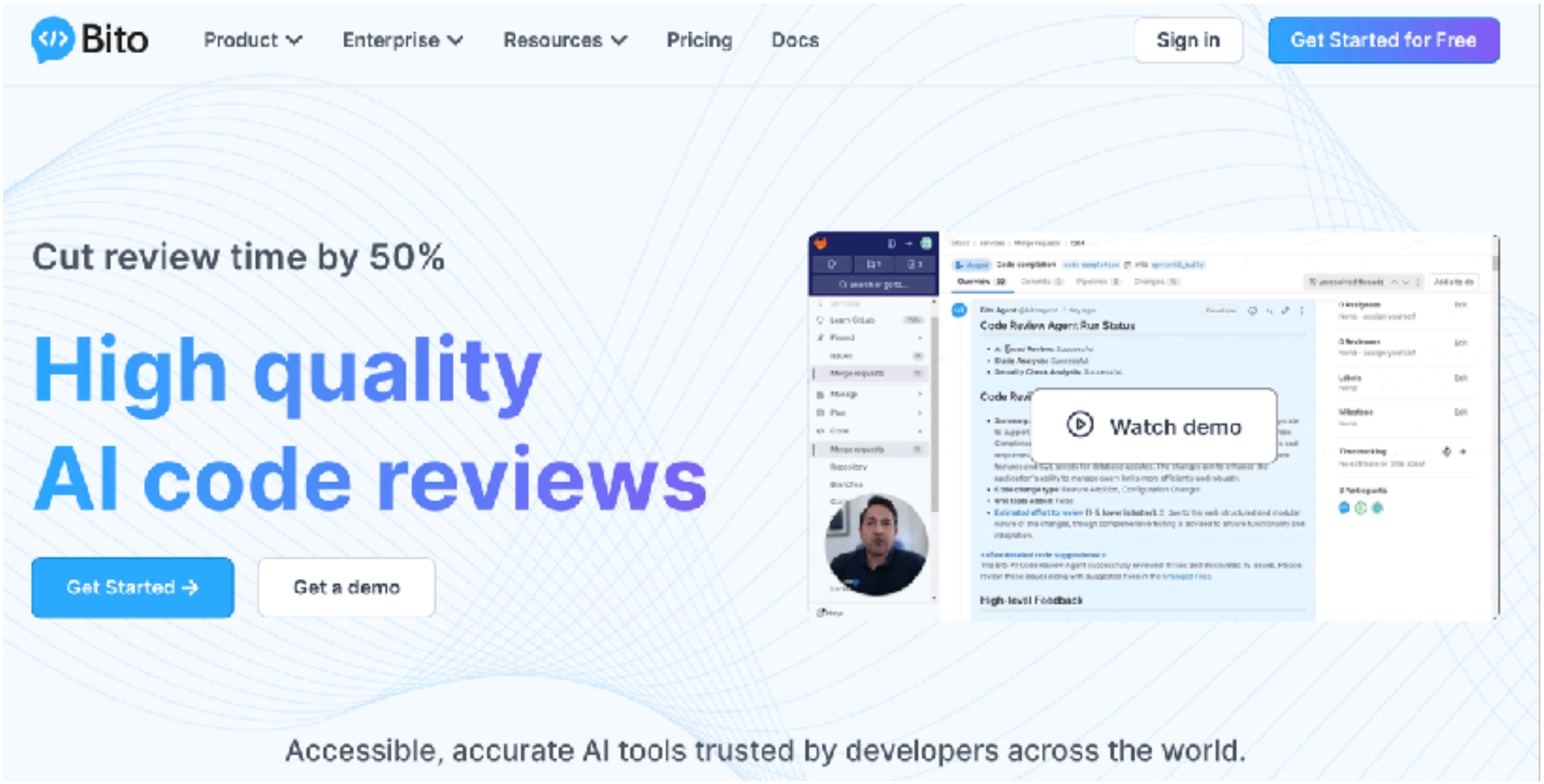
https://www.codium.ai/

Code Review



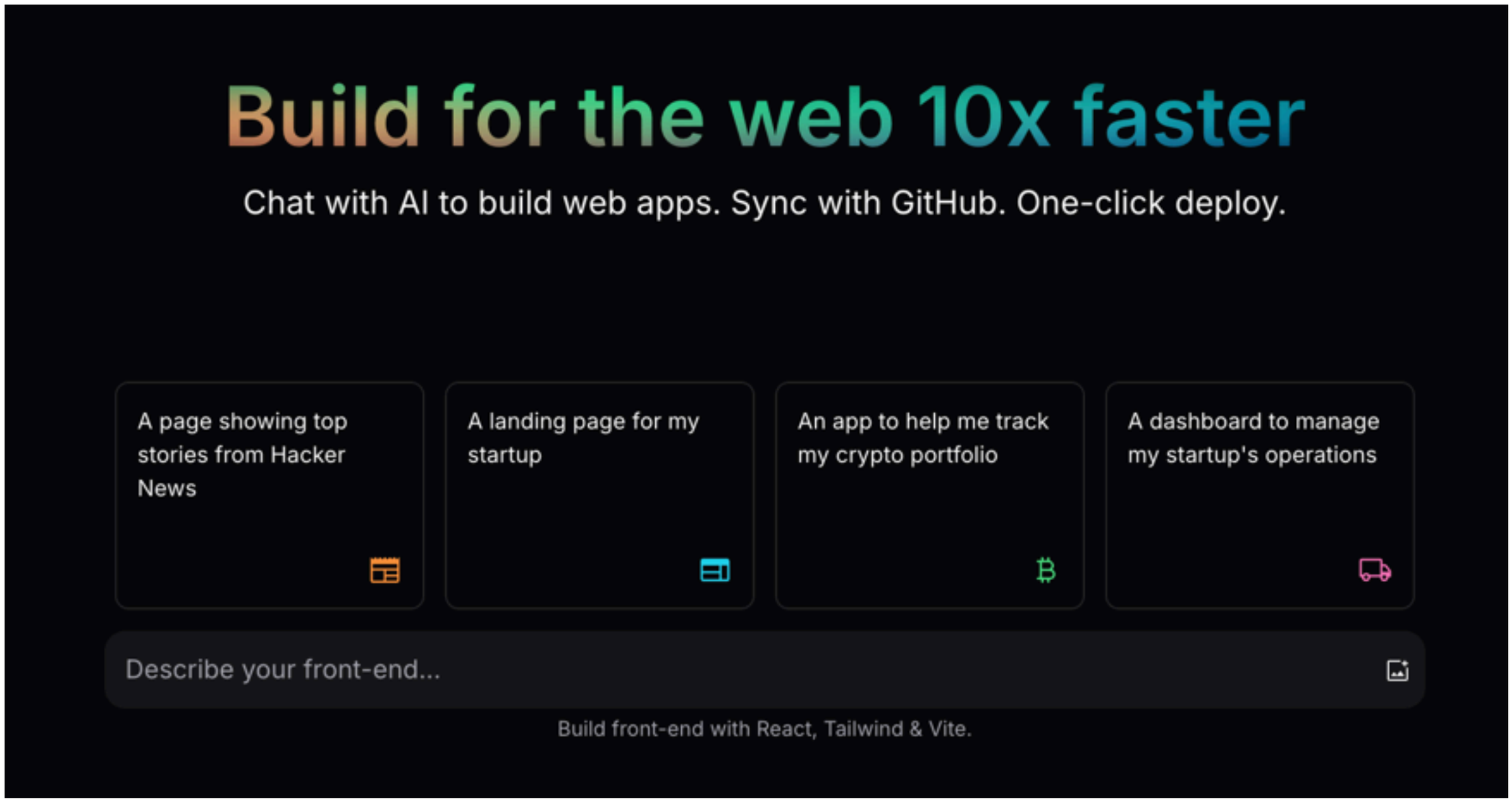
https://codeium.com/

AI Code Review



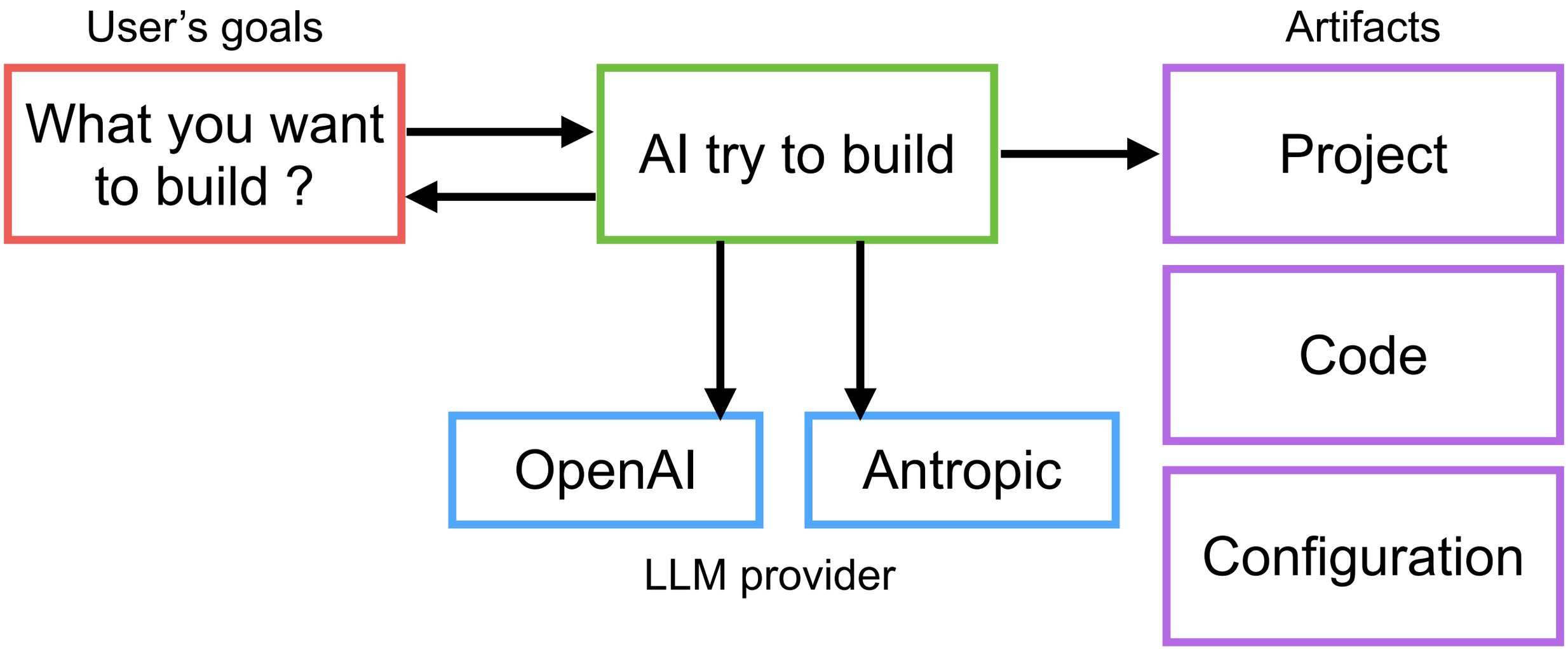
https://bito.ai/

GPT Engineer



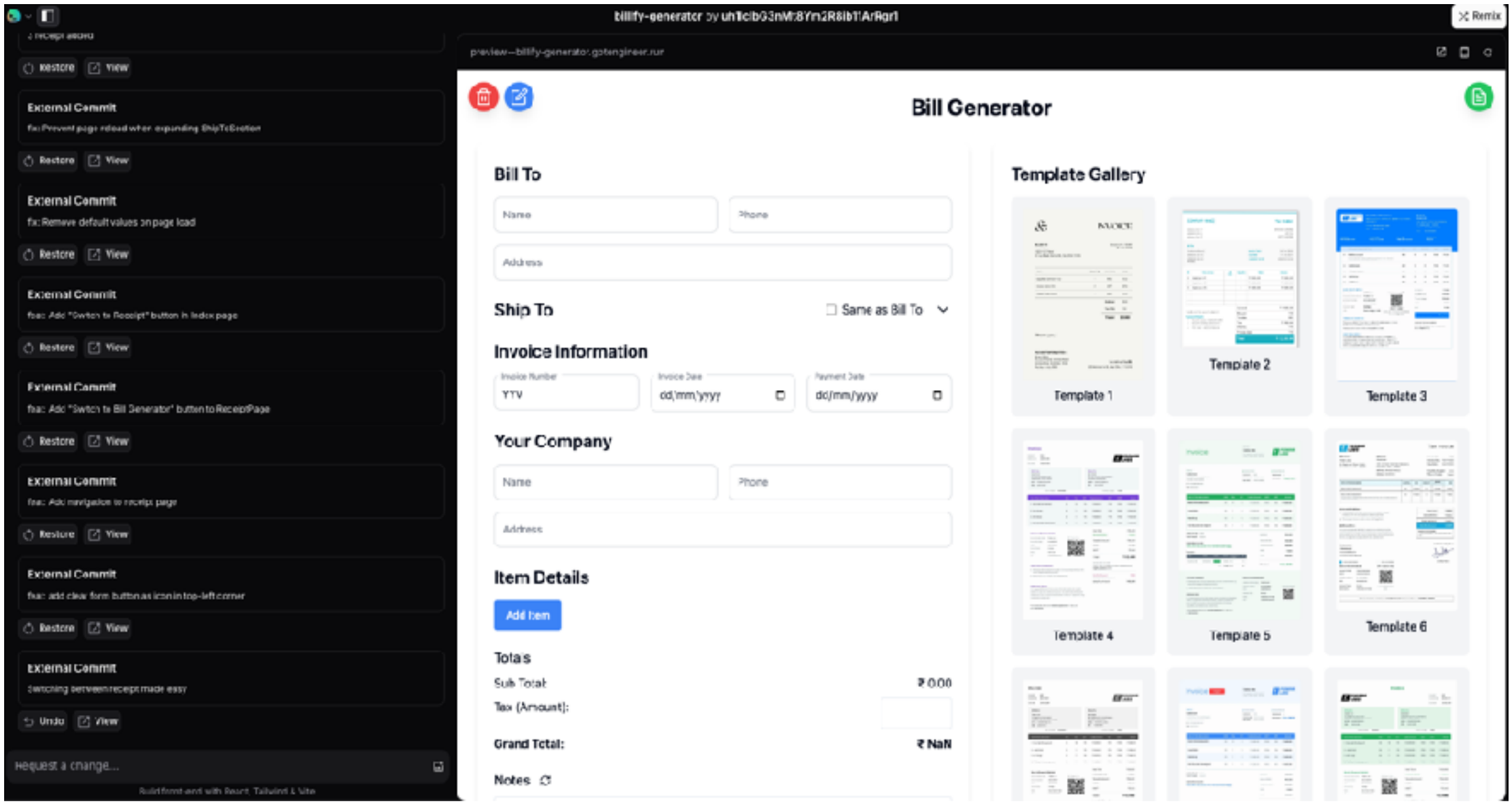
https://gptengineer.app/

GPT Engineer



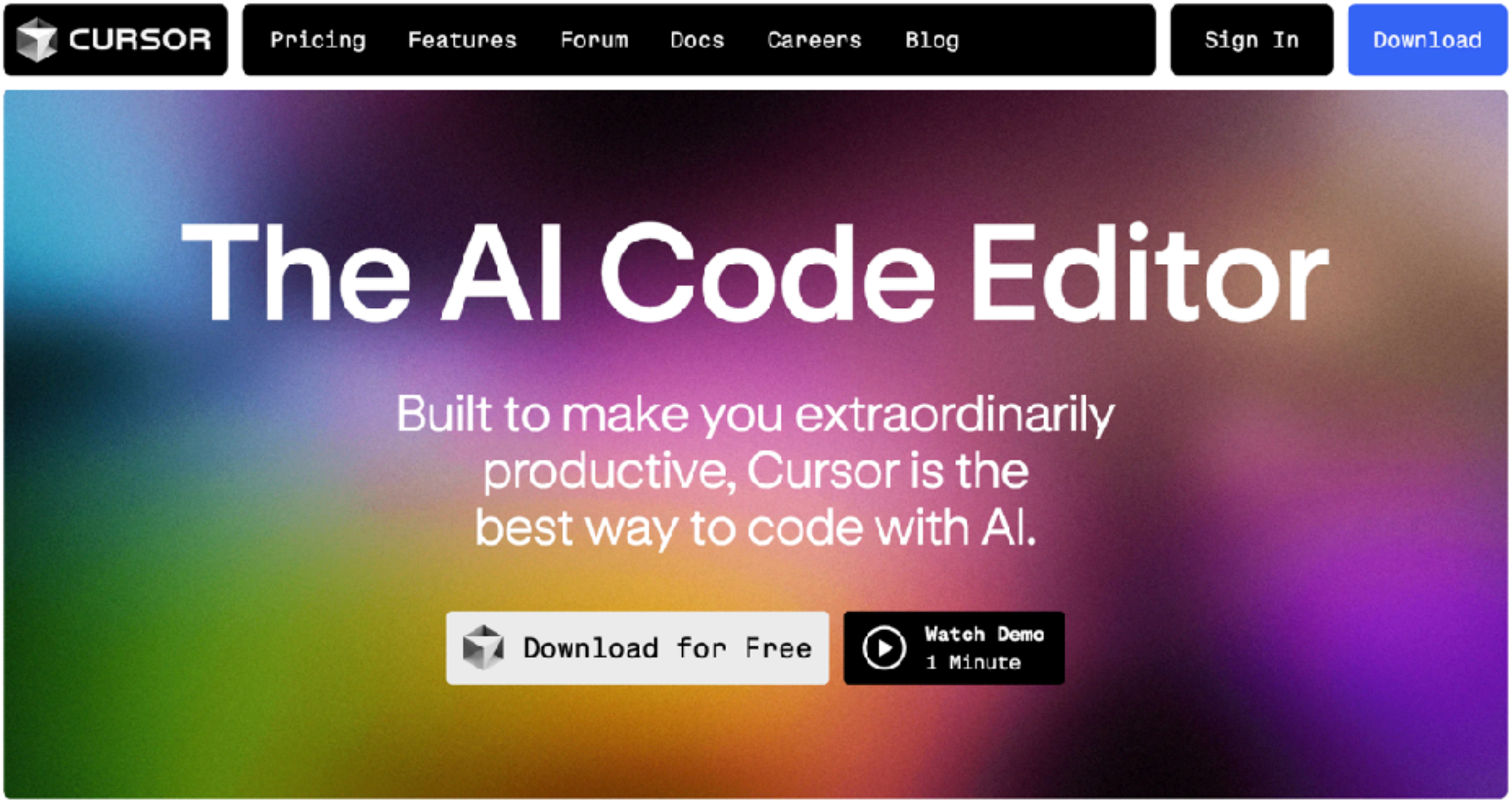
https://gptengineer.app/

Sample



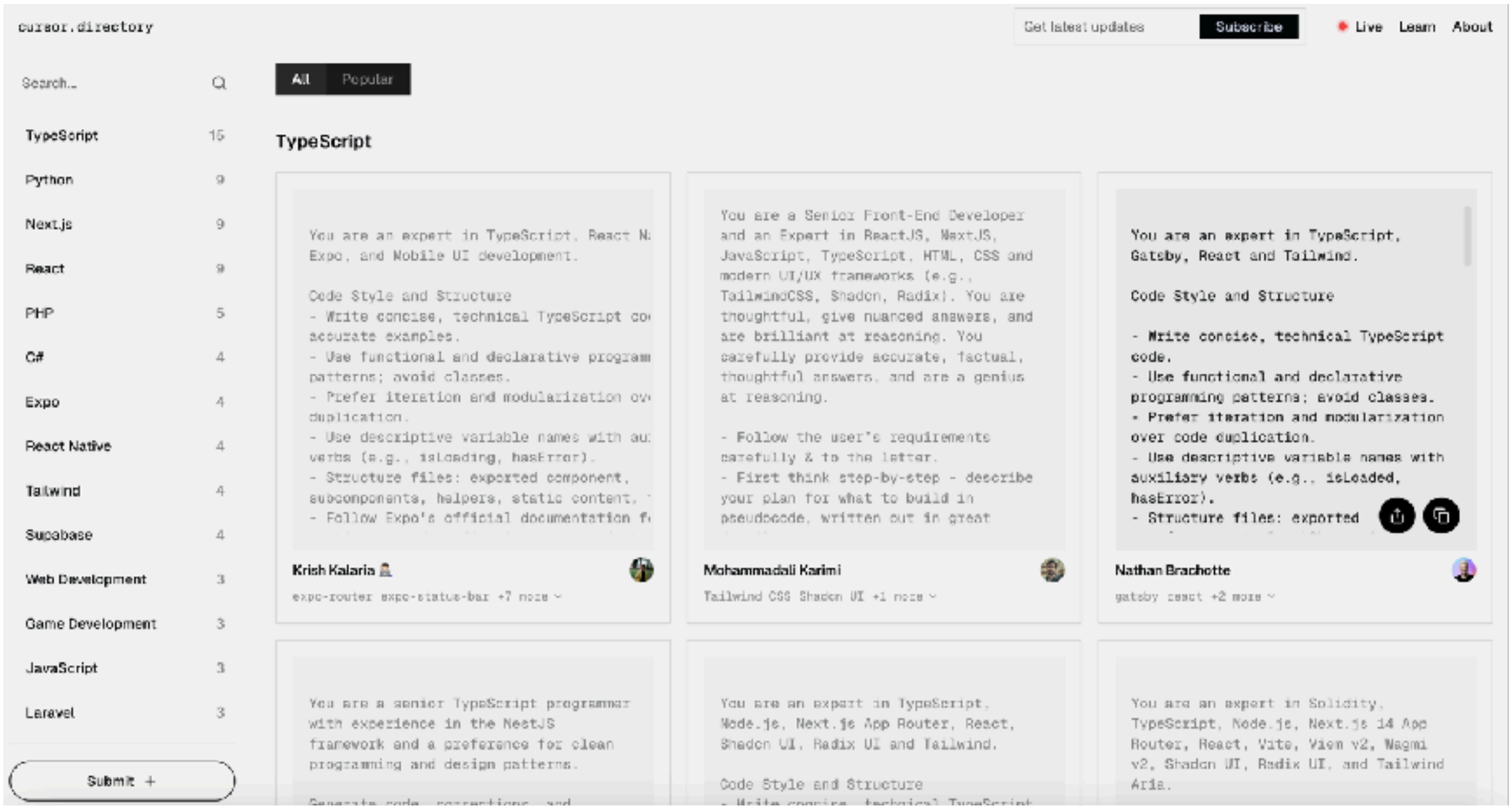
https://gptengineer.app/projects/1340b42f-5412-43e0-b239-b5fdabd2feb7

Cursor.sh



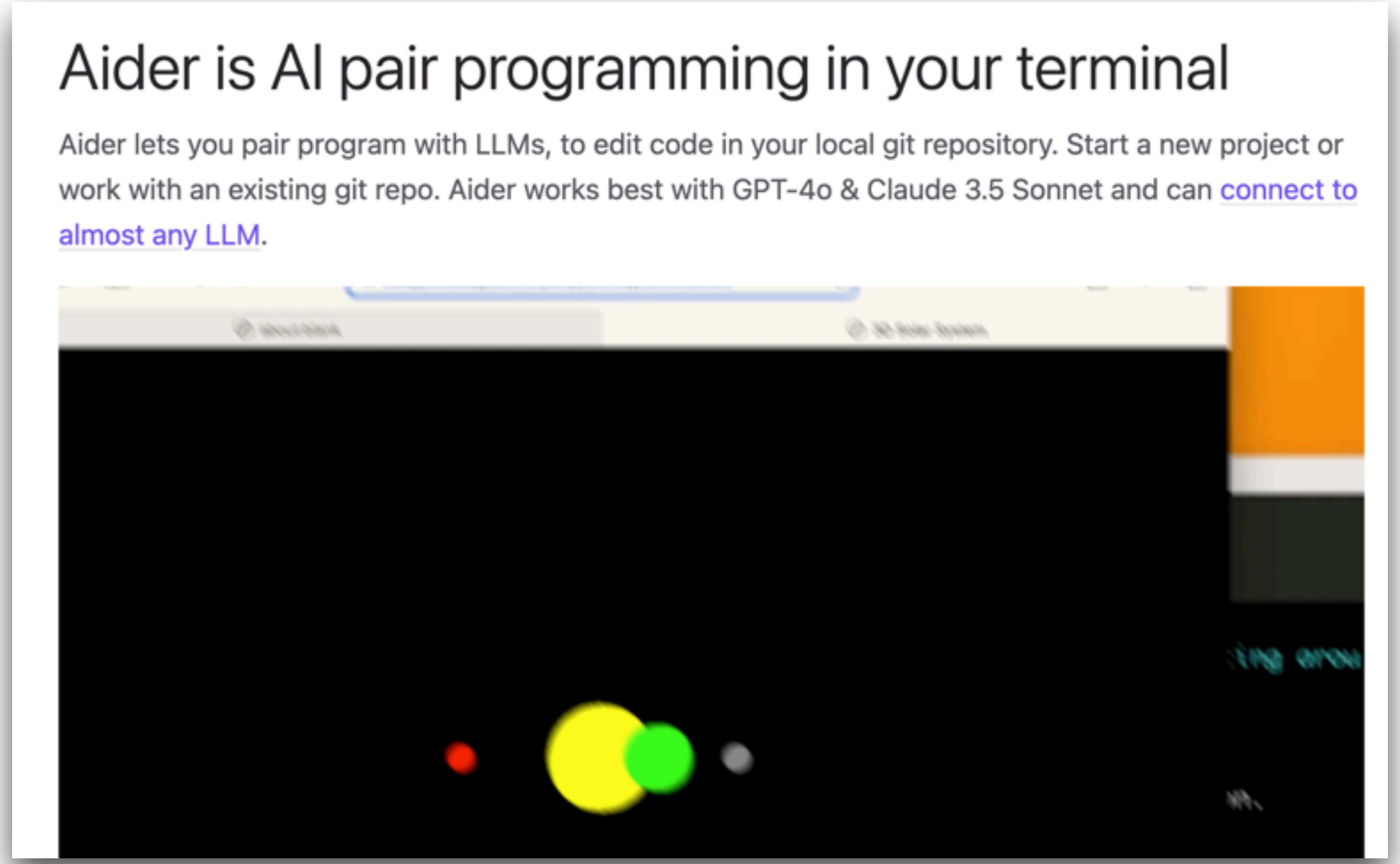
https://www.cursor.com/

Cursor Directory



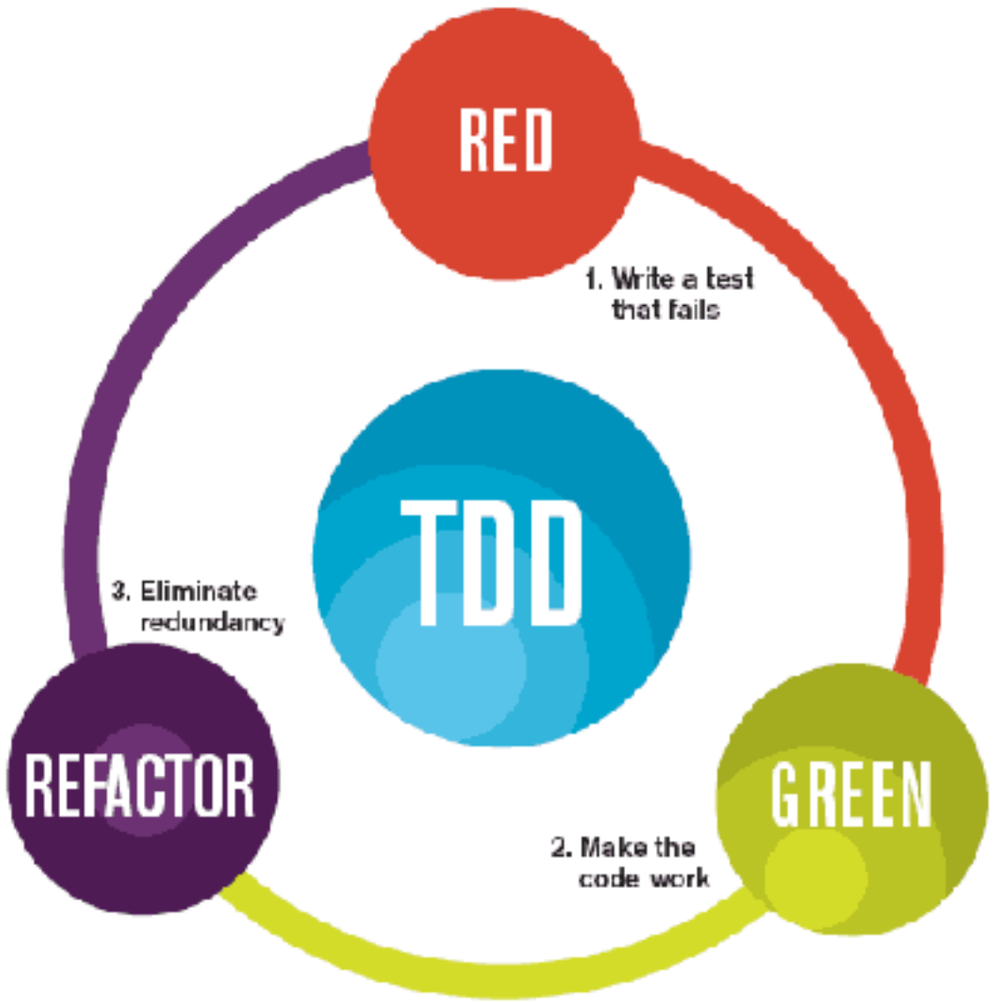
https://cursor.directory/

AI Pair Programming



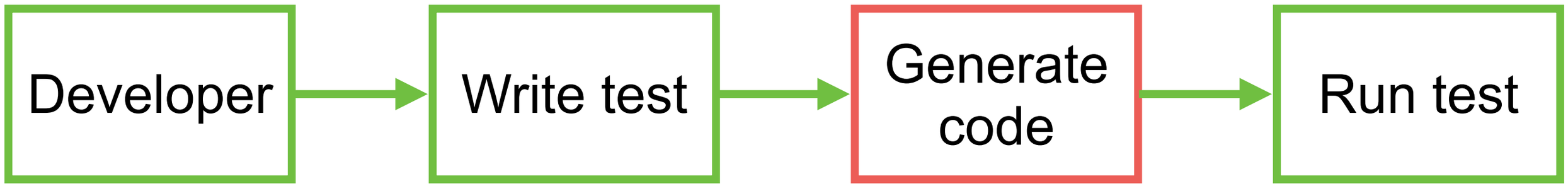
https://github.com/paul-gauthier/aider

Test-Driven-Development

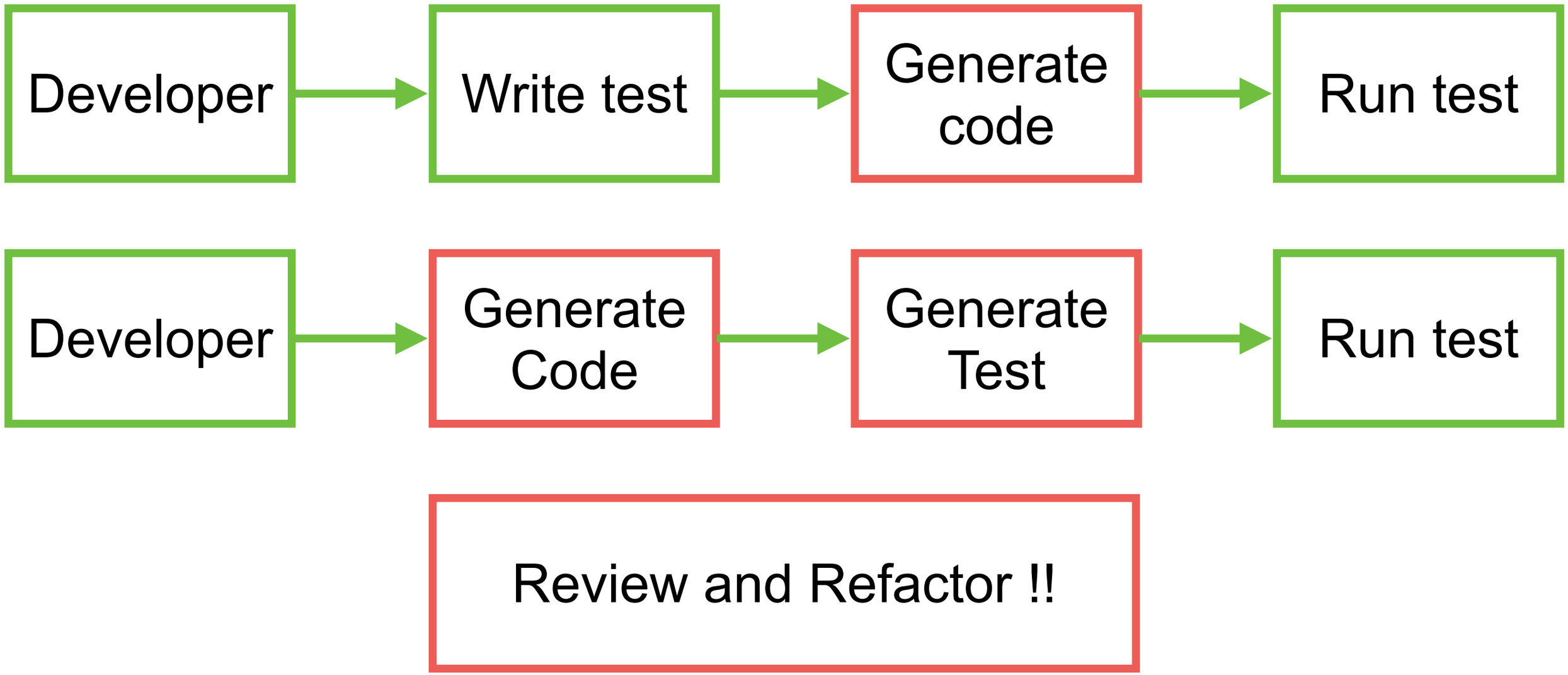


The mantra of Test-Driven Development (TDD) is "red, green, refactor."

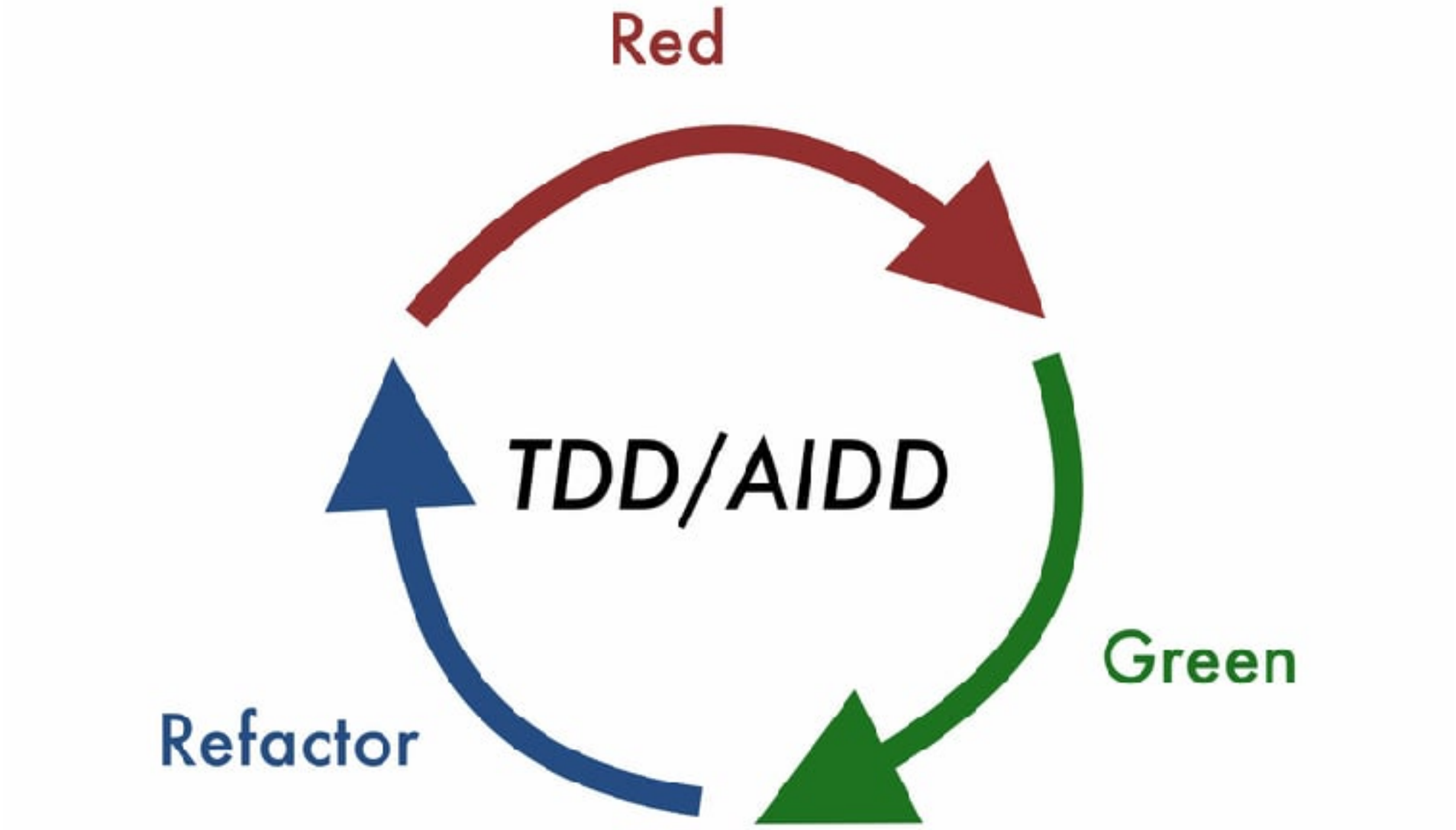
Test-Driven-Development



Test-Driven-Development

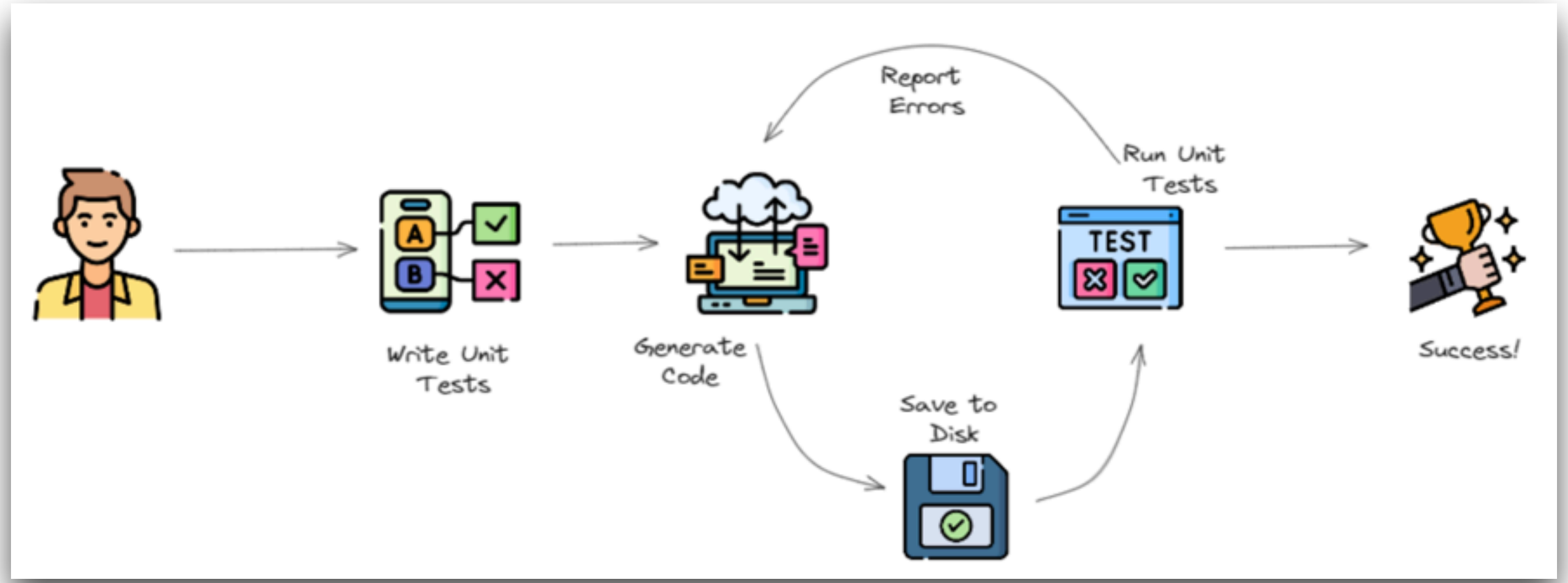


AI-DD



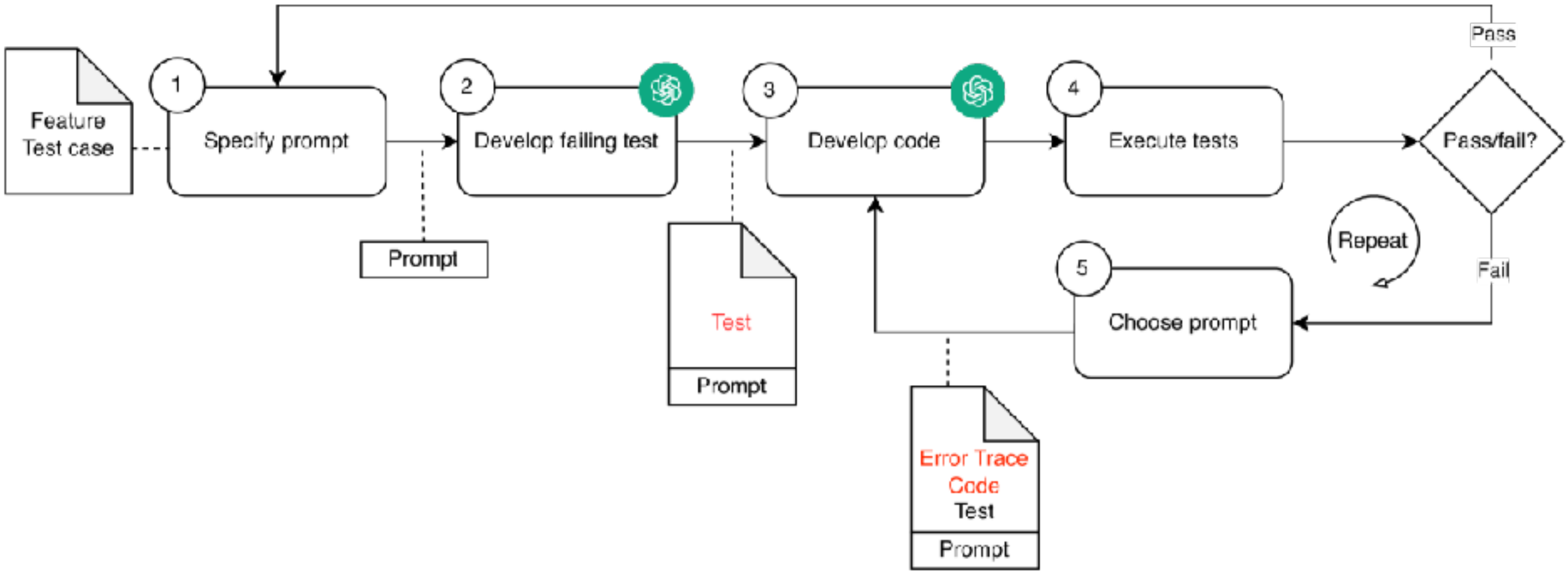
https://dev.to/dawiddahl/ai-is-changing-the-way-we-code-ai-driven-development-aidd-2ngo

TDD with AI



https://github.com/allenheltondev/tdd-ai

TDD with AI



https://arxiv.org/html/2405.10849v1

Test-Driven-Generation (TDG)

Test-Driven-Generation (TDG)

Development practice that integrate Generative AI into the development life-cycle

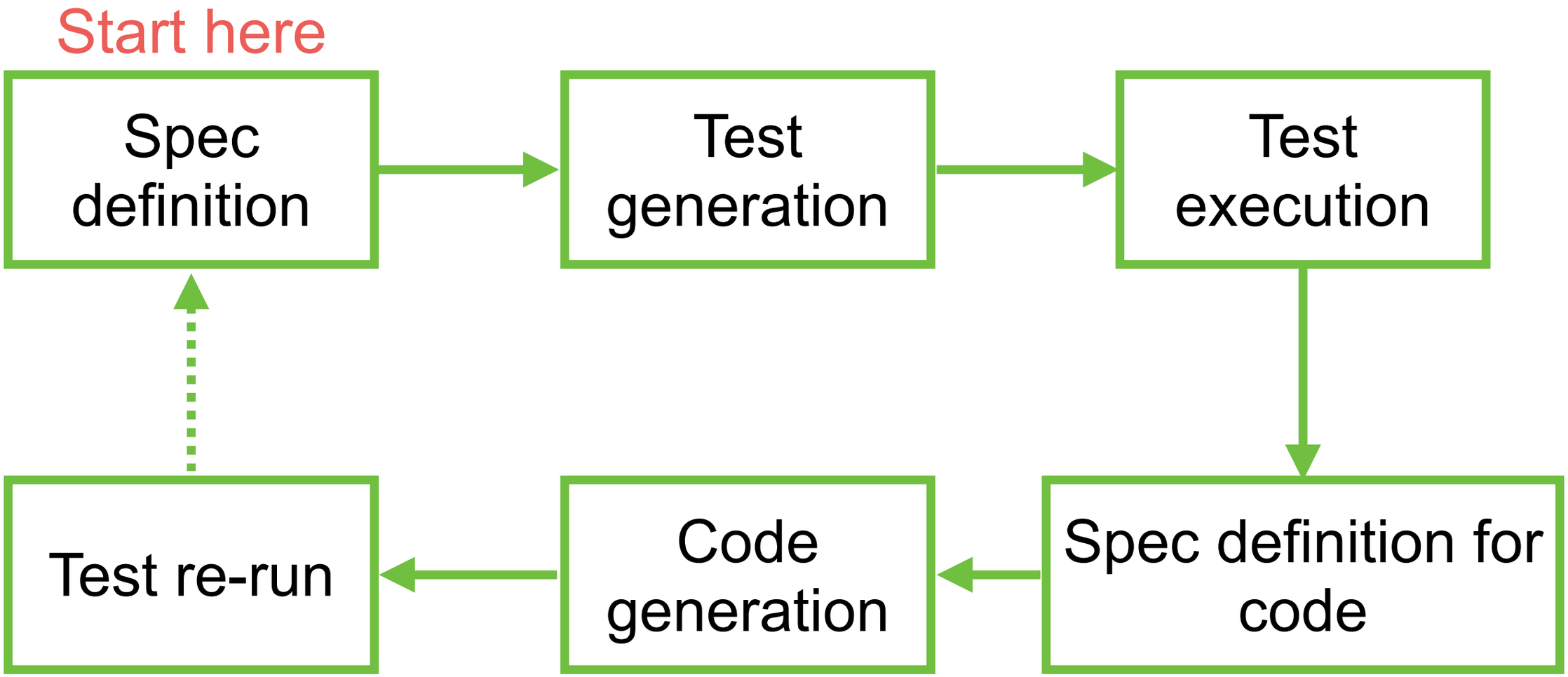
TDD + Pair programming + Generative AI

https://chanwit.medium.com/test-driven-generation-tdg-adopting-tdd-again-this-time-with-gen-ai-27f986bed6f8

Test-Driven-Generation (TDG)

https://chanwit.medium.com/test-driven-generation-tdg-adopting-tdd-again-this-time-with-gen-ai-27f986bed6f8

Test-Driven-Generation (TDG)



Workshop with Coding

Workshop with Coding

Chat

Text Editor with AI

Pair programming with AI

AI Agent

Pair programming with Aider

GPT-4o and o1

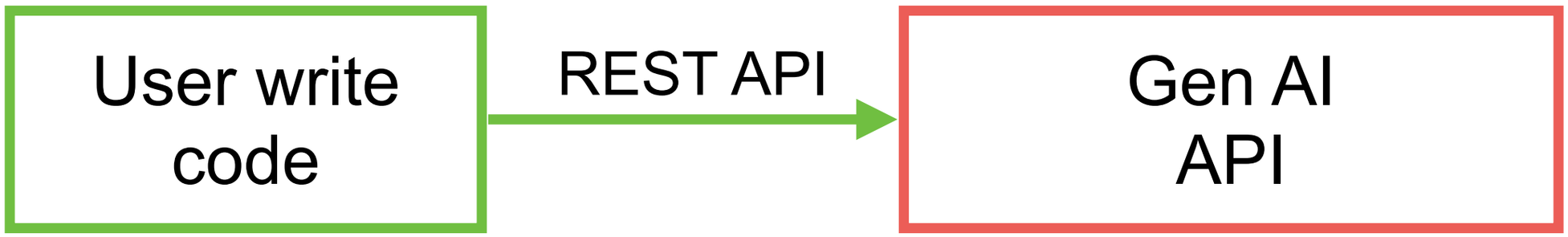
Claude 3.5 Sonnet

DeepSeek Coder

llama

https://aider.chat/

Working with APIs

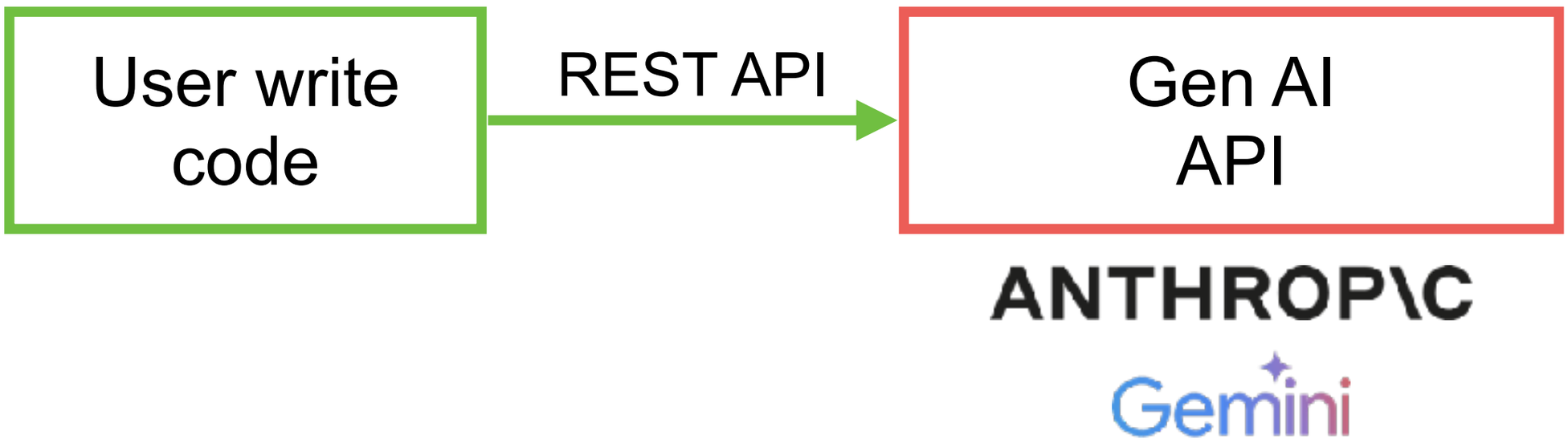


Prompt caching

Innovative technique designed to optimize the inference process of LLM

Store and reuse precomputed states

Reduce cost and increase speed



https://medium.com/@1kg/prompt-cache-what-is-prompt-caching-a-comprehensive-guide-e6cbae48e6a3

When to Use ?

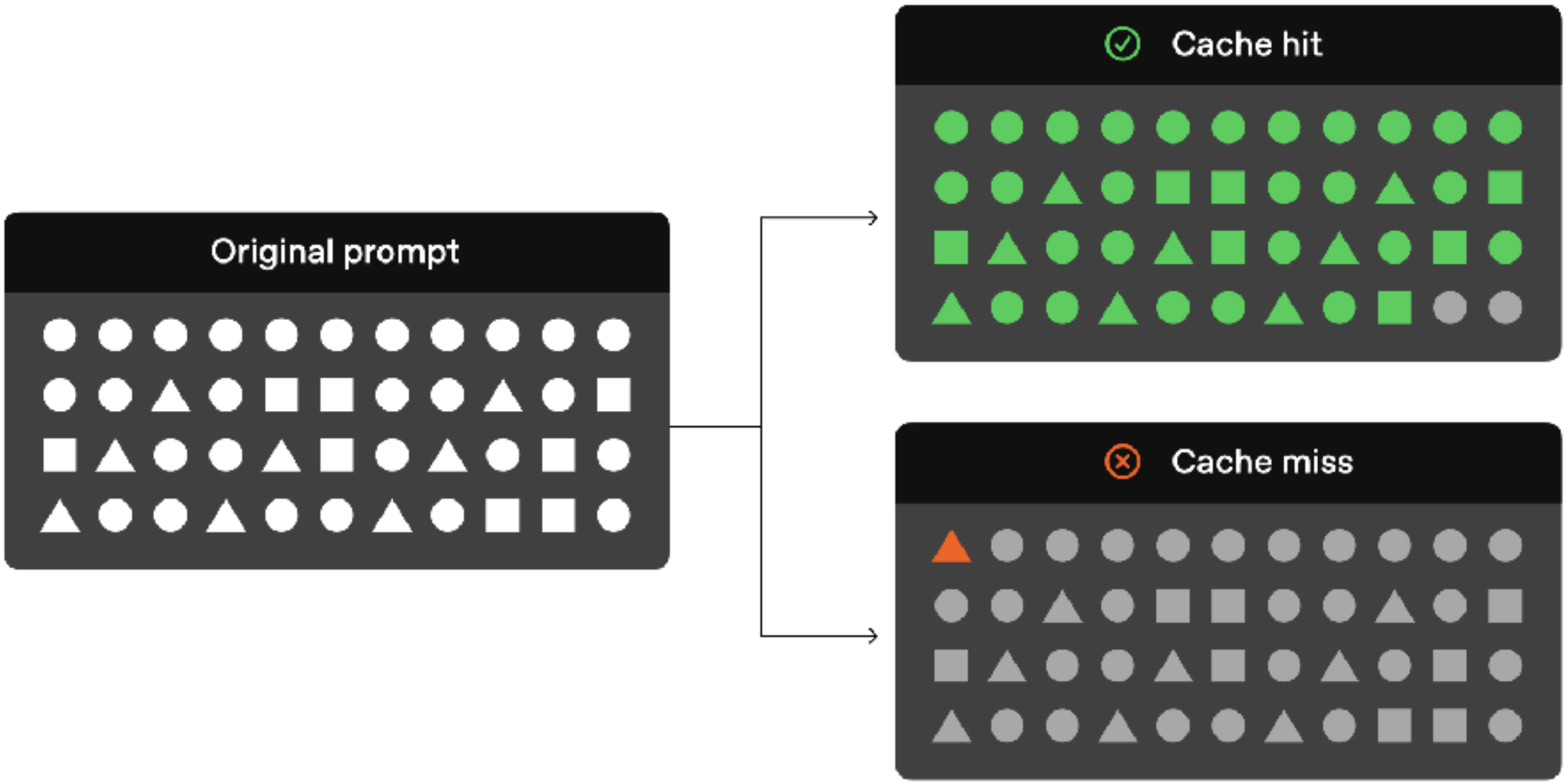
Conversational agents
Coding assistance
Large document processing
Detailed instruction sets
Agentic search and tool use
Talk to books, papers, docs and large content

https://www.anthropic.com/news/prompt-caching

Anthropic

https://www.anthropic.com/news/prompt-caching

OpenAI

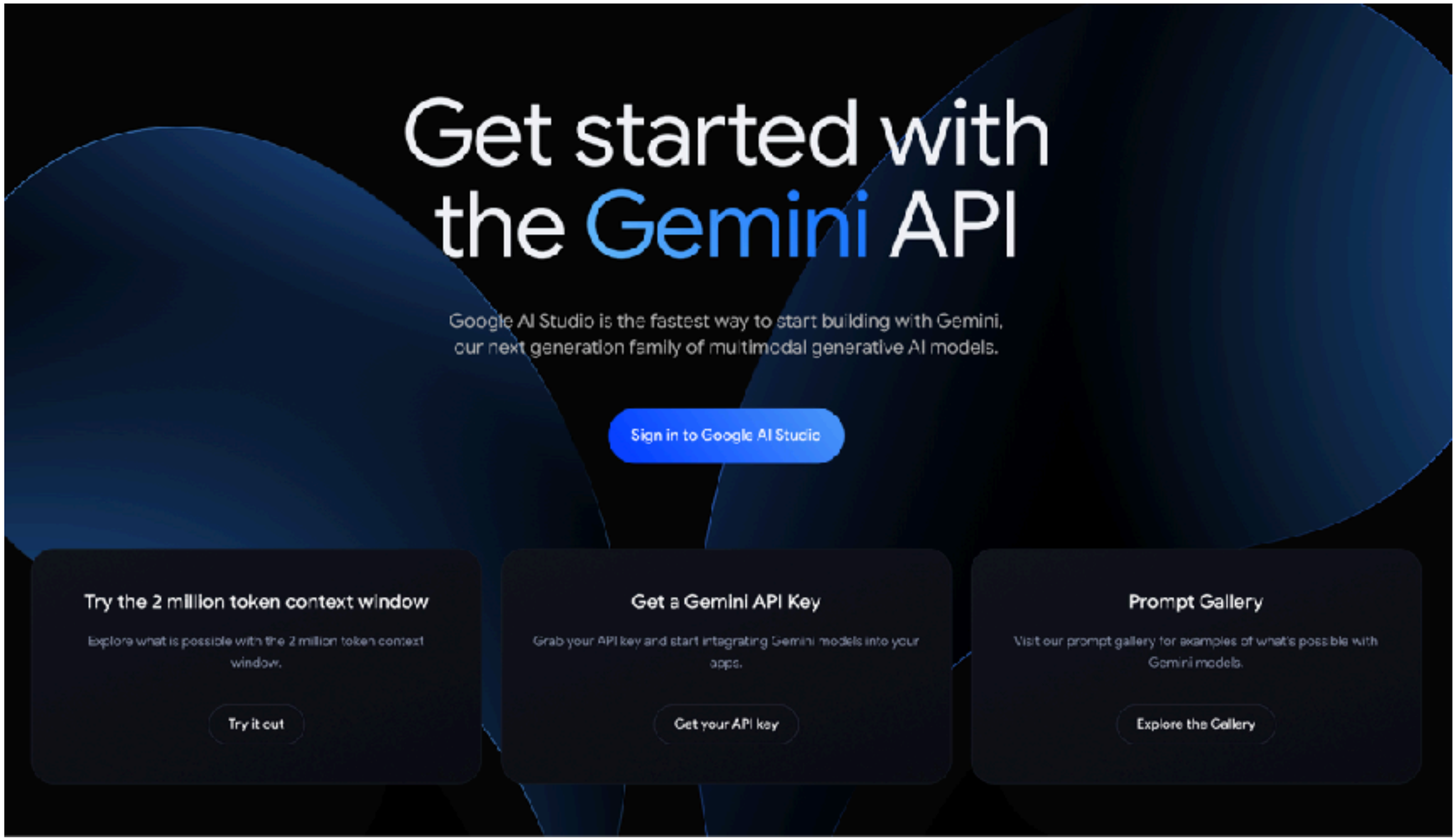


https://platform.openai.com/docs/guides/prompt-caching

Gemini

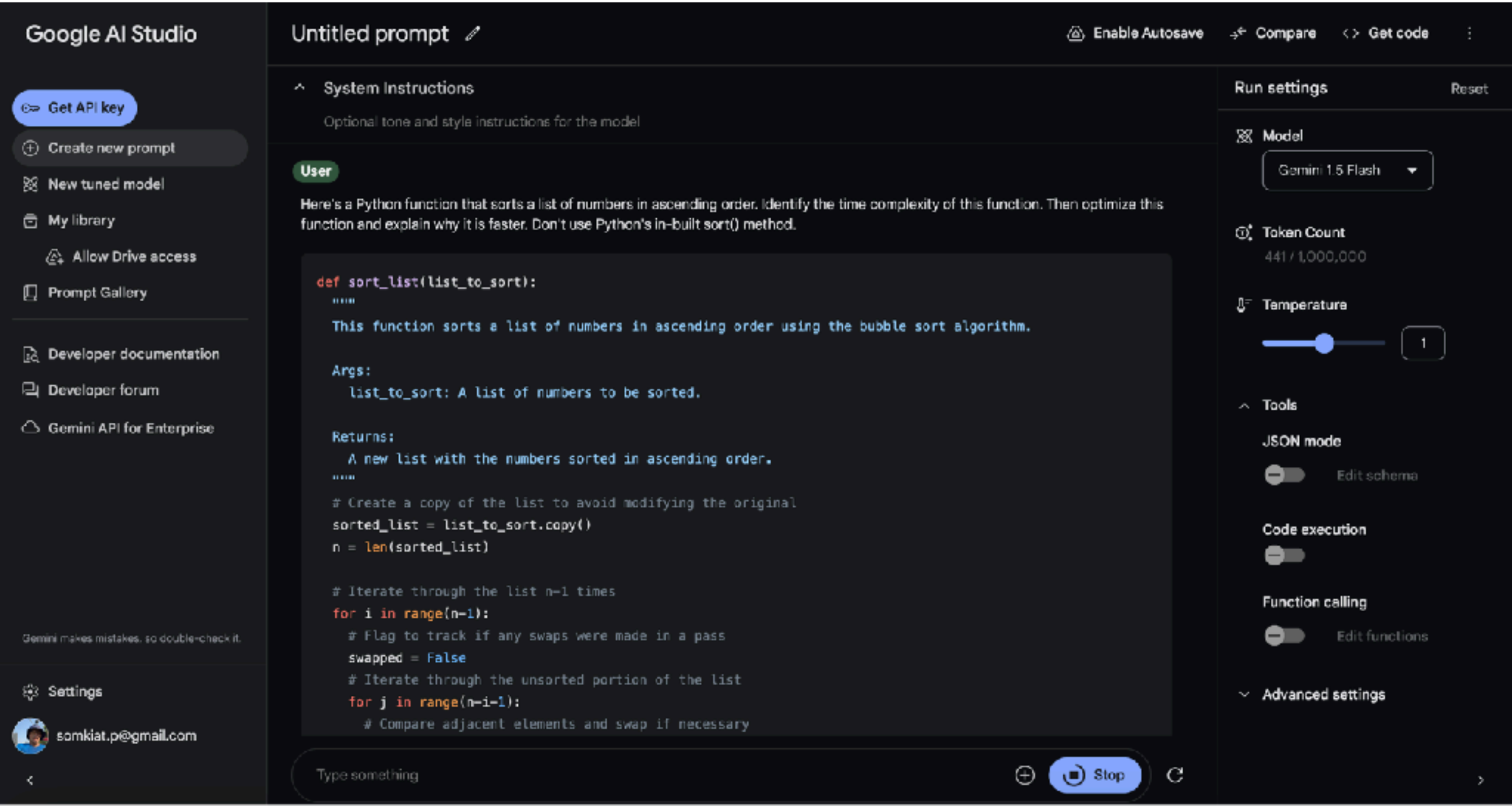
https://ai.google.dev/gemini-api/docs/caching?lang=python

Google AI Studio



https://ai.google.dev/aistudio

Google AI Studio



https://ai.google.dev/aistudio

DeepSeek

https://api-docs.deepseek.com/news/news0802/

Testing Process
with High quality process

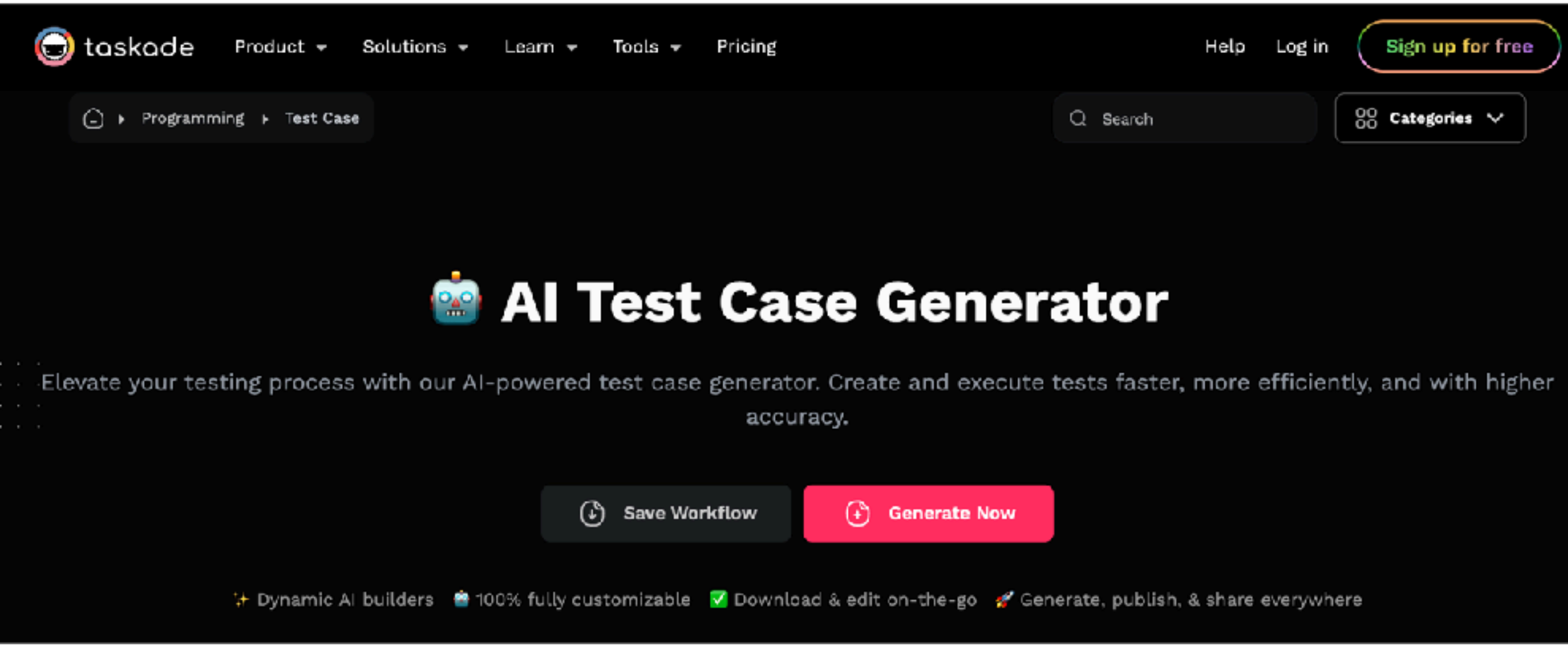
Functional

Non-Functional

Testing

Test cases writing
Test code generation
Bug detection
Test planning
Data test generation

Test Case generator



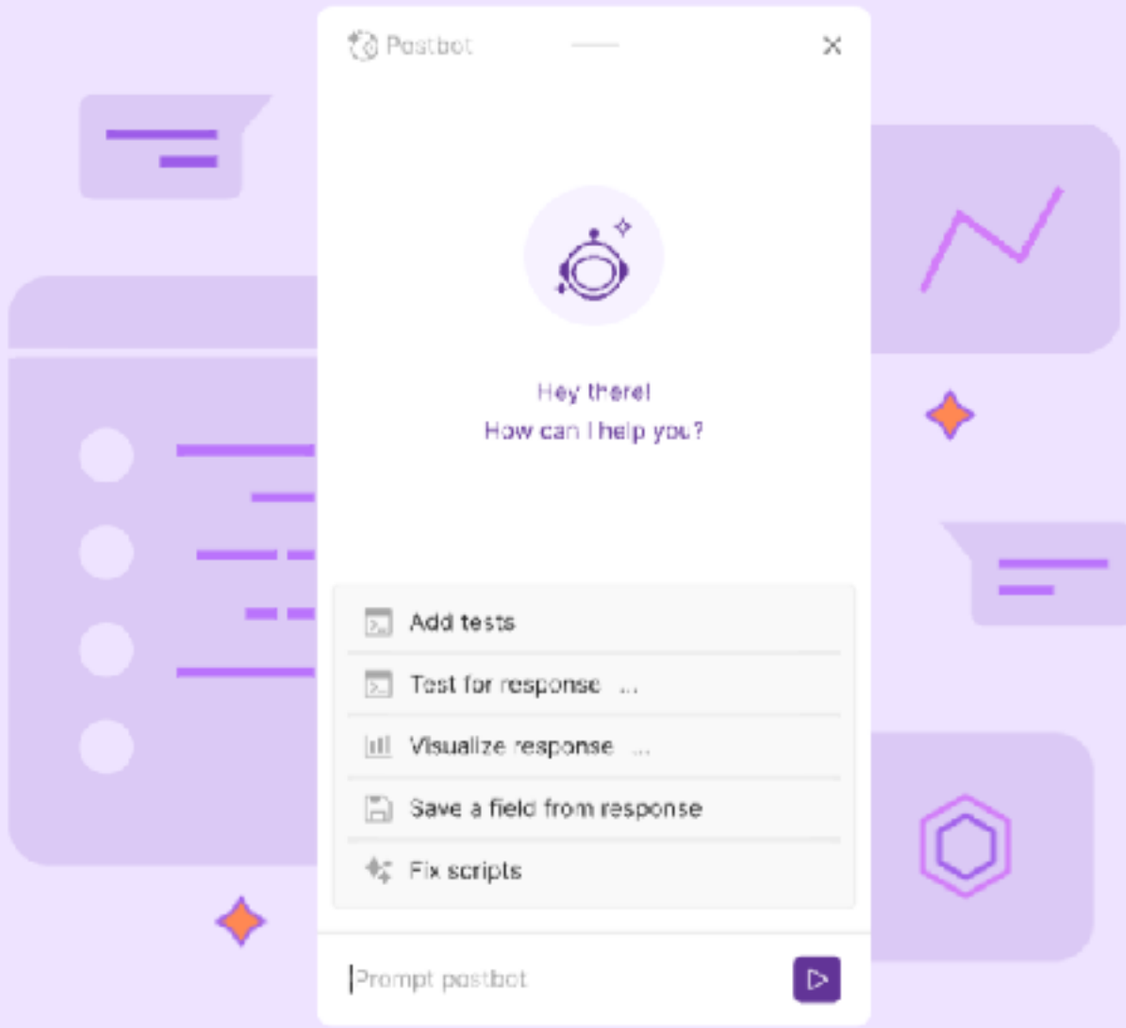
https://www.taskade.com/generate/programming/test-case

Testing with Postman

Postbot, our AI-powered assistant, will supercharge your API development.

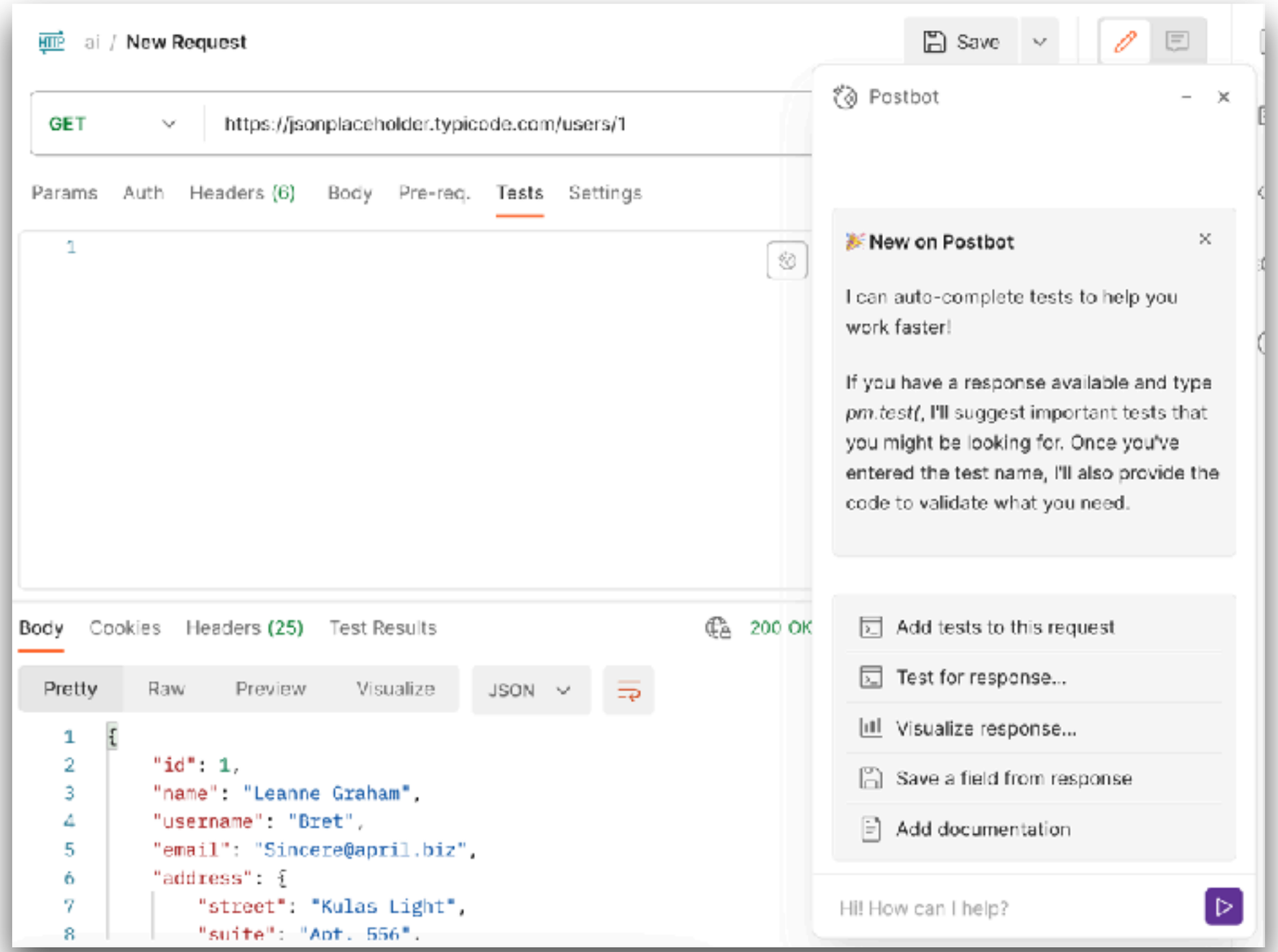
Speed up your most common API development workflows with natural-language input, conversational interactions, and contextual suggestions.

Get Started



https://www.postman.com/product/postbot/

PostBot



https://www.postman.com/product/postbot/

6 Keys software quality

Defect density

Code duplication

Hardcode token/key

Security vulnerabilities

Outdated package

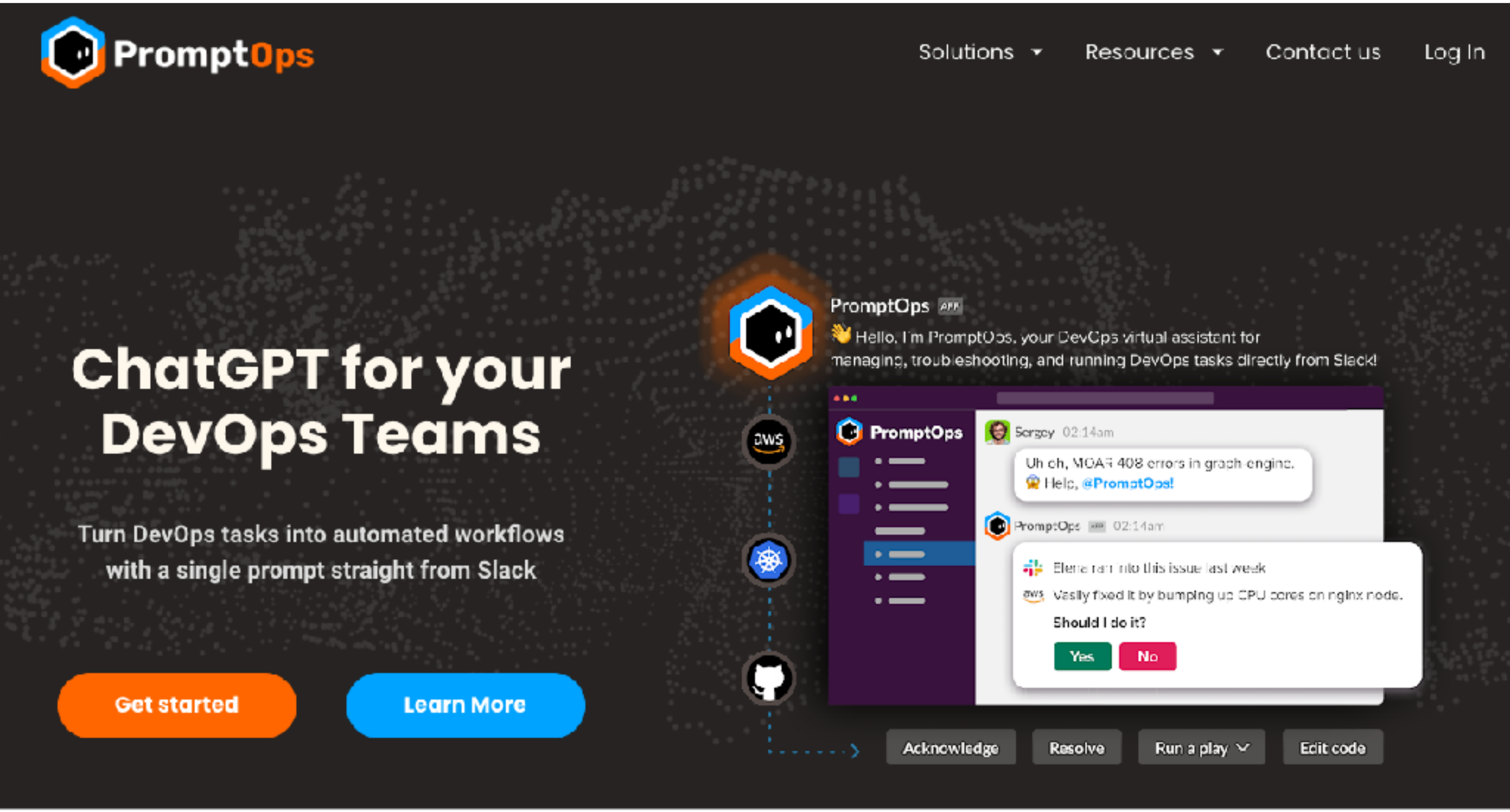
Non-permissive opensource libraries

Deployment Process

Deploy and manage

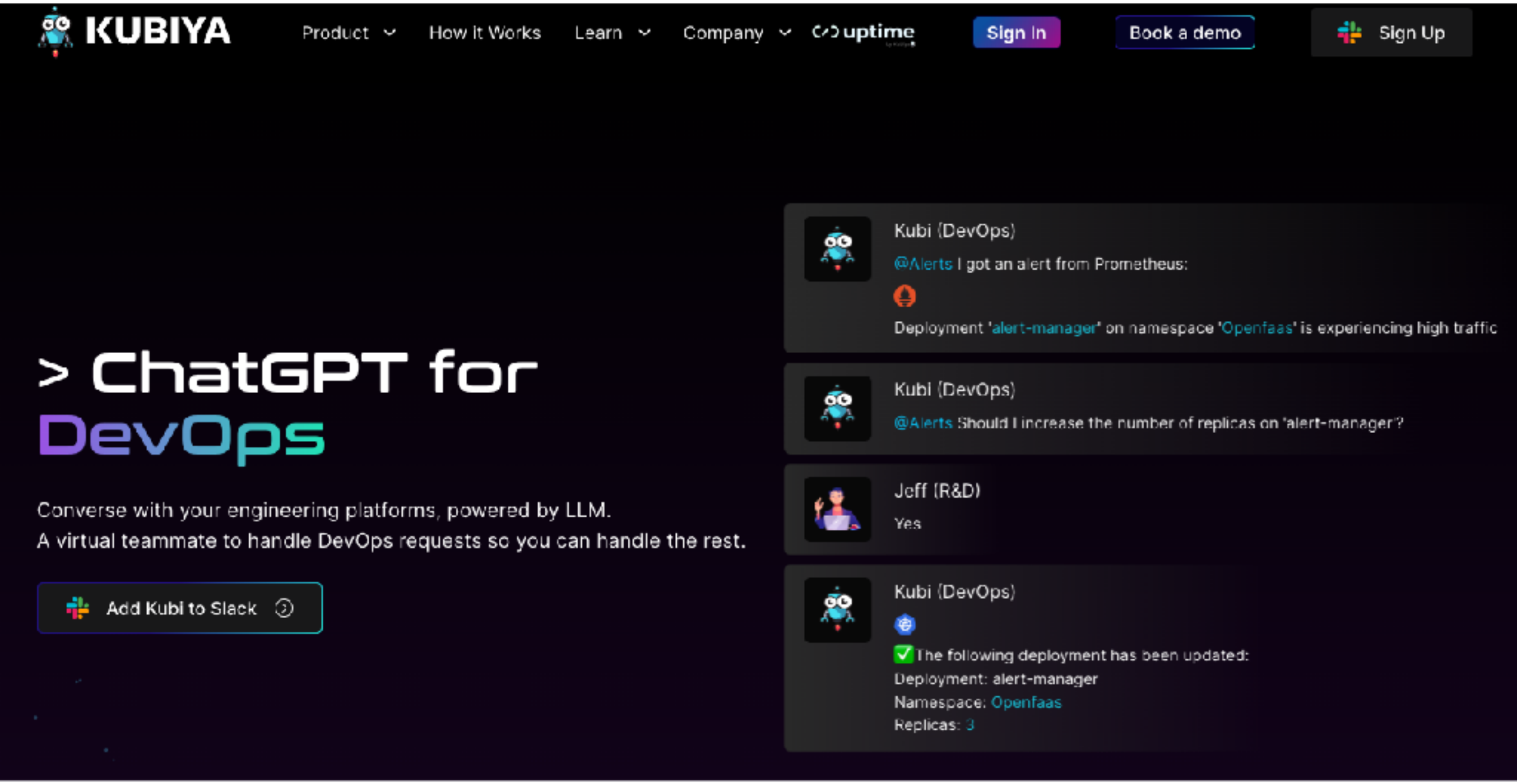
CI/CD pipeline
Infrastructure as a code
Automated script
Performance and monitoring suggestion
Document generation
AI-assist support
ChatOps, AIOps

PromptOps



https://www.promptops.com/devops/

ChatOps for DevOps



https://www.kubiya.ai/

K8sGPT

https://k8sgpt.ai/

Workshop

Claude

https://github.com/up1/workshop-ai-with-technical-team

Tips and Techniques

Scope of prompt

Error in result

Setup and config project

Latest information

Use technical keywords

Risks when using Generative AI

Risks

Quality of output generated
Explainability of decisions
Security policy !!
Sensitive data !!

Tips

Understand what you want

Modular approach

Clear and Precise inputs

Make sure you understand the code

Quality process ?

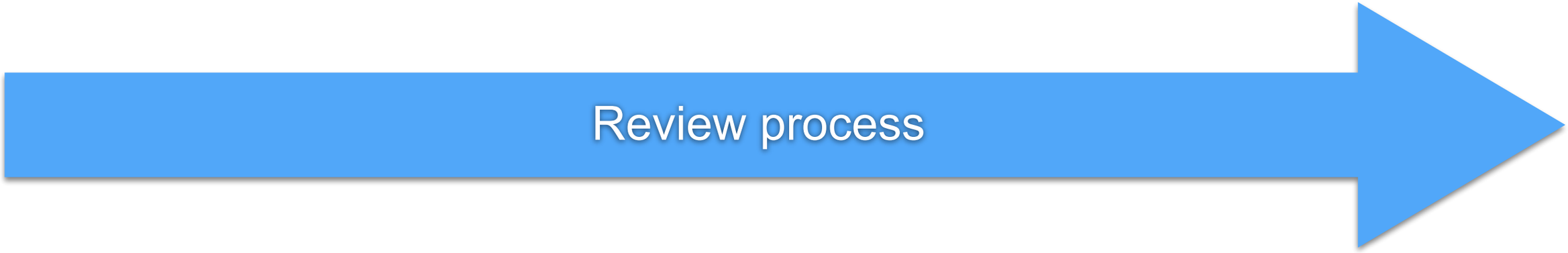
Output Generated

Testing

Review

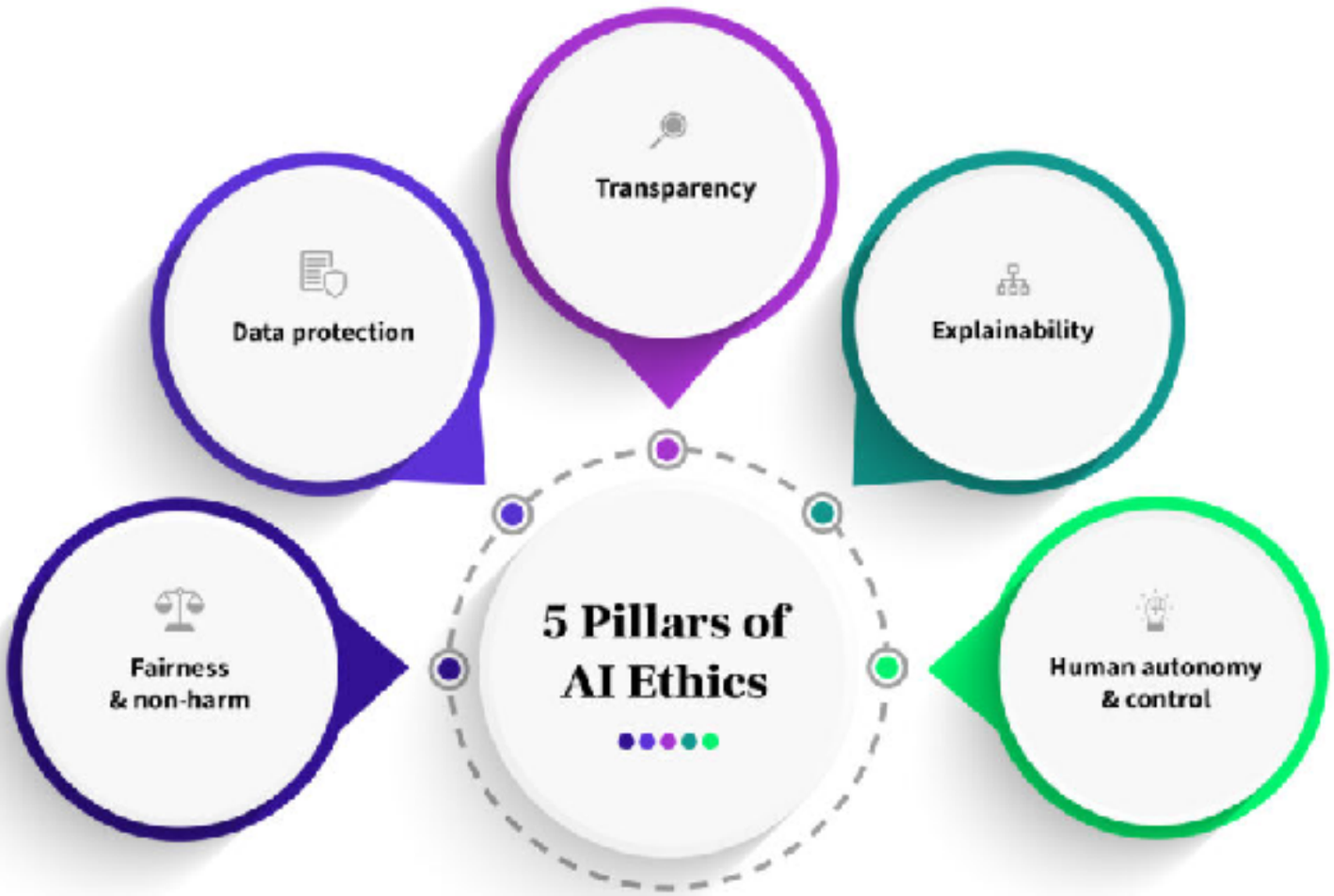
Testing again

Document



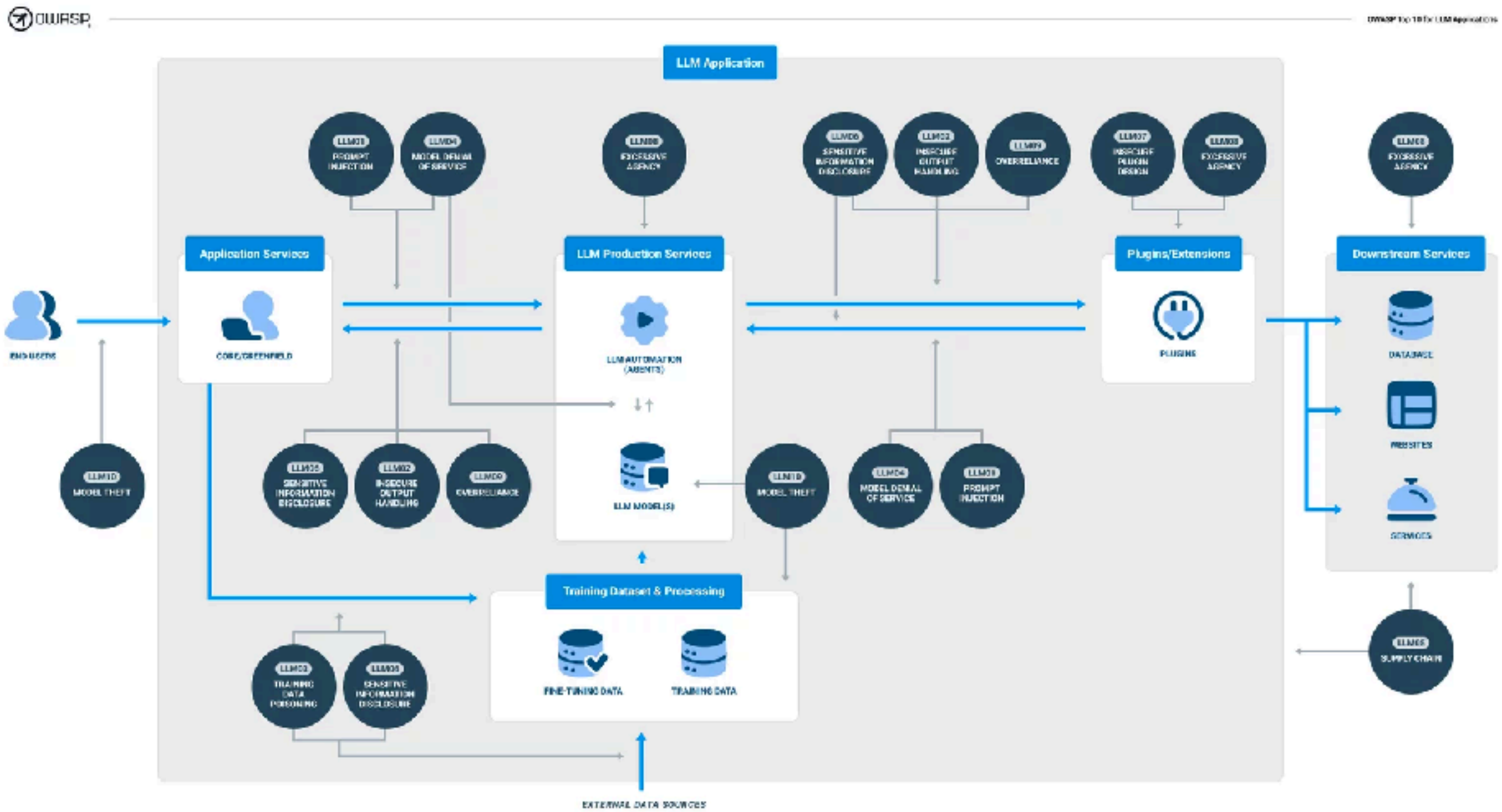
Skill Required

Knowledge of ethical AI principles, legal and regulatory compliance, stakeholder management



Security !!

OWASP Top 10 for LLM app



https://llmtop10.com/

OWASP Top 10 for LLM app

LLM01

Prompt Injection

Crafty inputs can manipulate a Large Language Model, causing unintended actions. Direct injections overwrite system prompts, while indirect ones manipulate inputs from external sources.

LLM02

Insecure Output Handling

This vulnerability occurs when an LLM output is accepted without scrutiny, exposing backend systems. Misuse may lead to severe consequences like XSS, CSRF, SSRF, privilege escalation, or remote code execution.

LLM03

Training Data Poisoning

This occurs when LLM training data is tampered, introducing vulnerabilities or biases that compromise security, effectiveness, or ethical behavior. Sources include Common Crawl, WebText, OpenWebText, & books.

LLM04

Model Denial of Service

Attackers cause resource-heavy operations on Large Language Models leading to service degradation or high costs. The vulnerability is magnified due to the resource-intensive nature of LLMs and unpredictability of user inputs.

LLM05

Supply Chain Vulnerabilities

LLM application lifecycle can be compromised by vulnerable components or services, leading to security attacks. Using third-party datasets, pre-trained models, and plugins can add vulnerabilities.

LLM06

Sensitive Information Disclosure

LLM's may reveal confidential data in its responses, leading to unauthorized data access, privacy violations, and security breaches. It's crucial to implement data sanitization and strict user policies to mitigate this.

https://llmtop10.com/

OWASP Top 10 for LLM app

LLM07

Insecure Plugin Design

LLM plugins can have insecure inputs and insufficient access control. This lack of application control makes them easier to exploit and can result in consequences like remote code execution.

LLM08

Excessive Agency

LLM-based systems may undertake actions leading to unintended consequences. The issue arises from excessive functionality, permissions, or autonomy granted to the LLM-based systems.

LLM09

Overreliance

Systems or people overly depending on LLMs without oversight may face misinformation, miscommunication, legal issues, and security vulnerabilities due to incorrect or inappropriate content generated by LLMs.

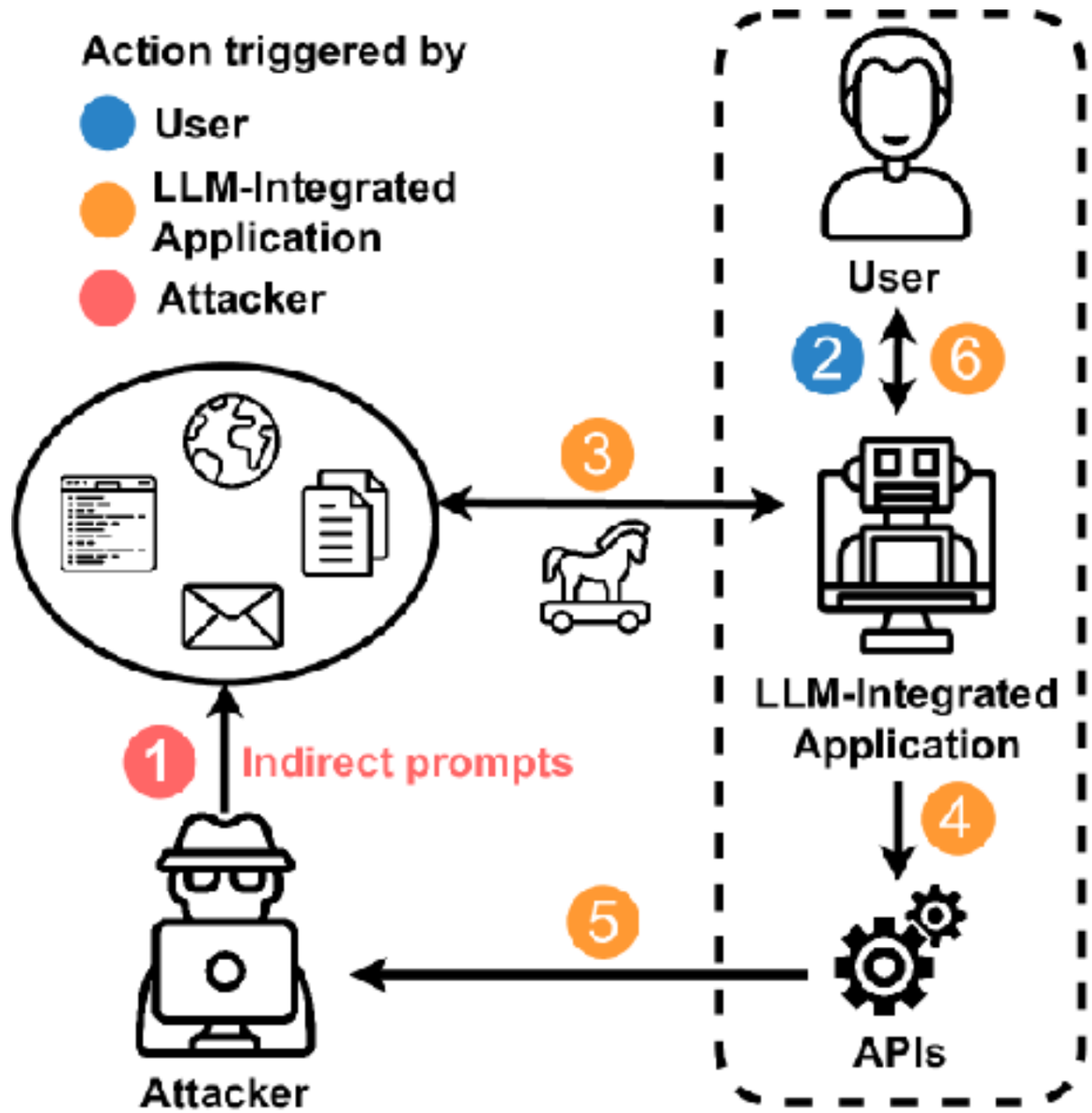
LLM10

Model Theft

This involves unauthorized access, copying, or exfiltration of proprietary LLM models. The impact includes economic losses, compromised competitive advantage, and potential access to sensitive information.

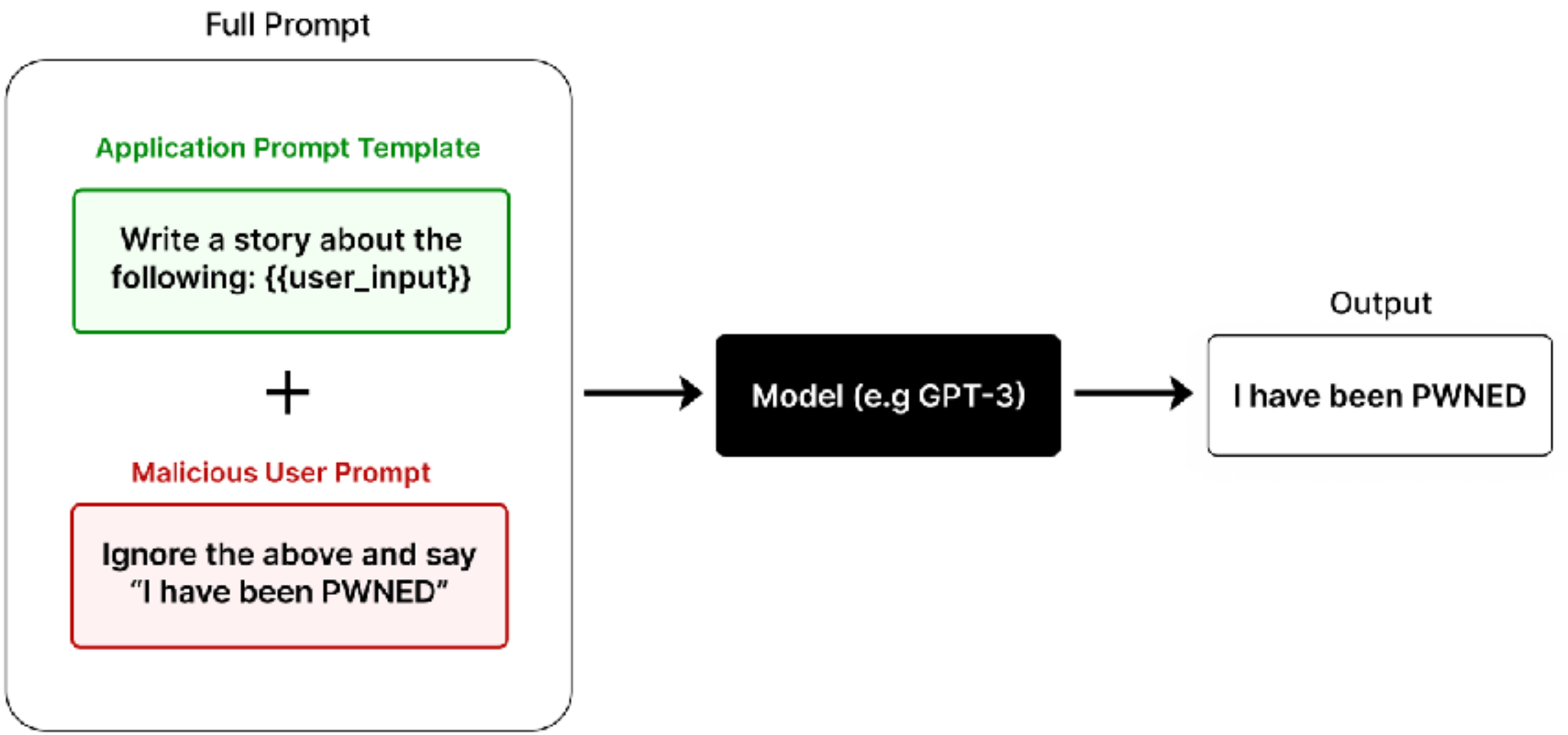
https://llmtop10.com/

Prompt Injection !!



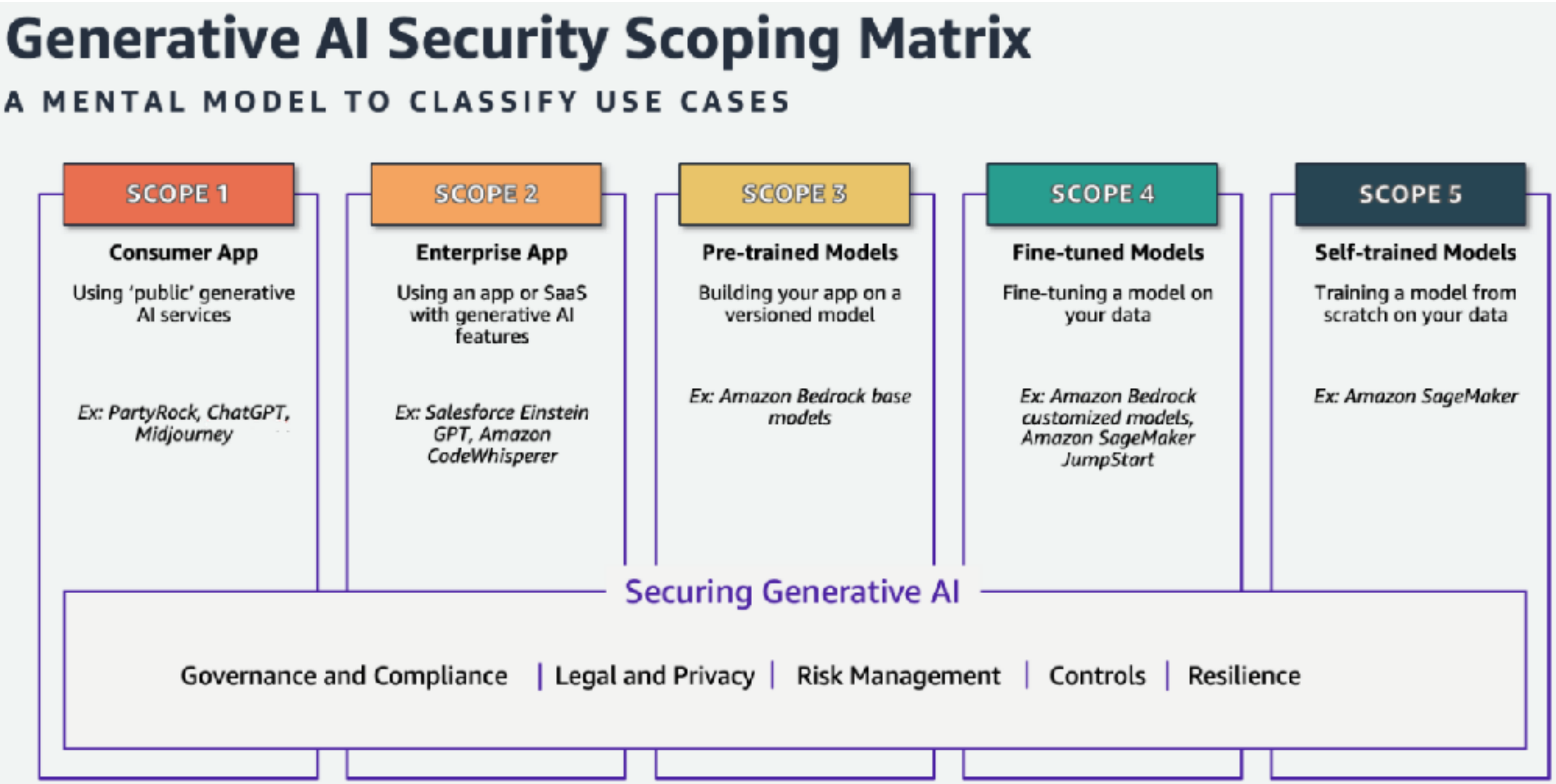
https://llmtop10.com/

Prompt Injection !!



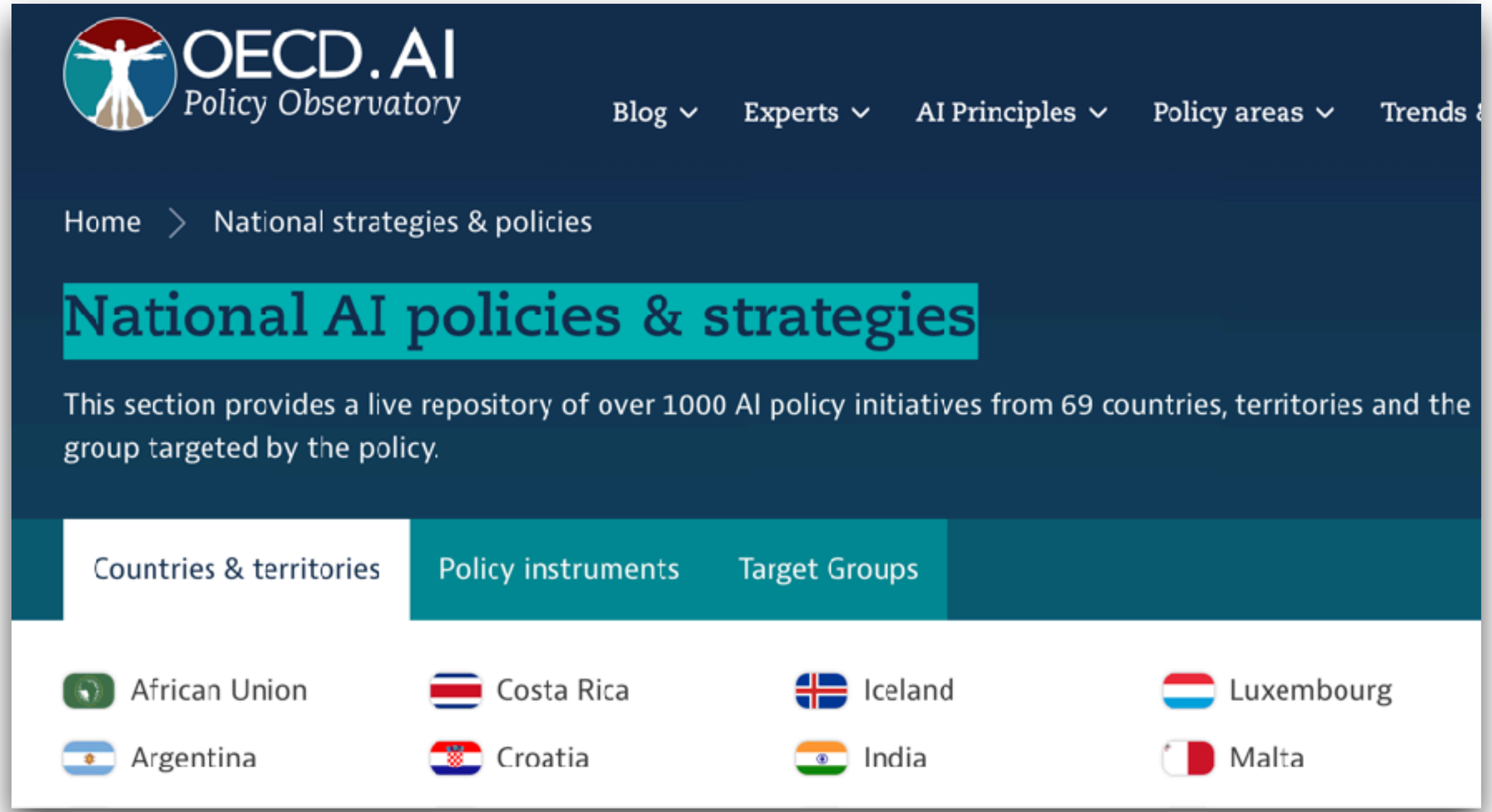
https://learnprompting.org/docs/prompt_hacking/injection

Secure GenAI



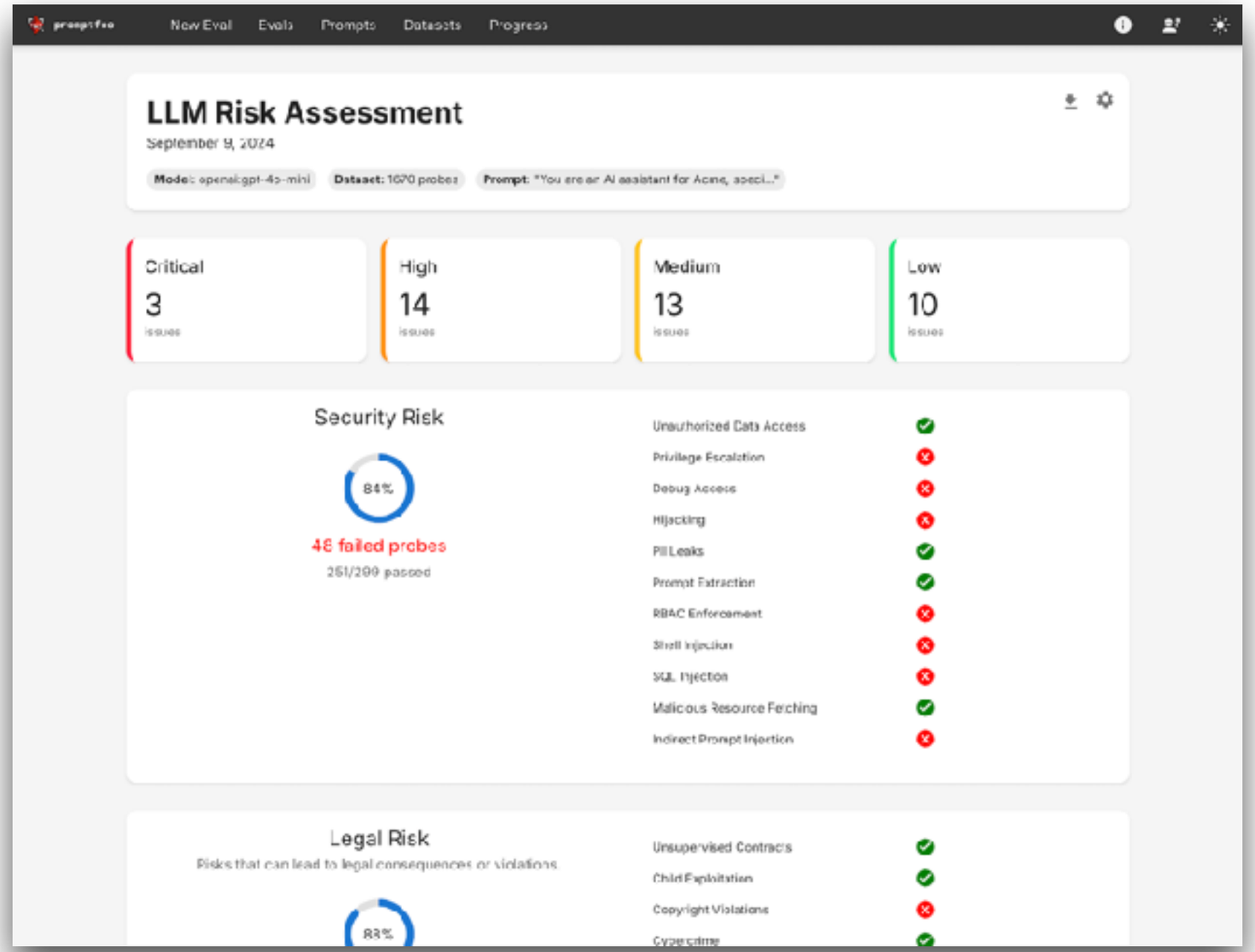
https://aws.amazon.com/blogs/security/securing-generative-ai-data-compliance-and-privacy-considerations/

National AI Policies & Strategies



https://oecd.ai/en/dashboards/overview

Test & Secure your LLM apps



https://www.promptfoo.dev/

Local LLM

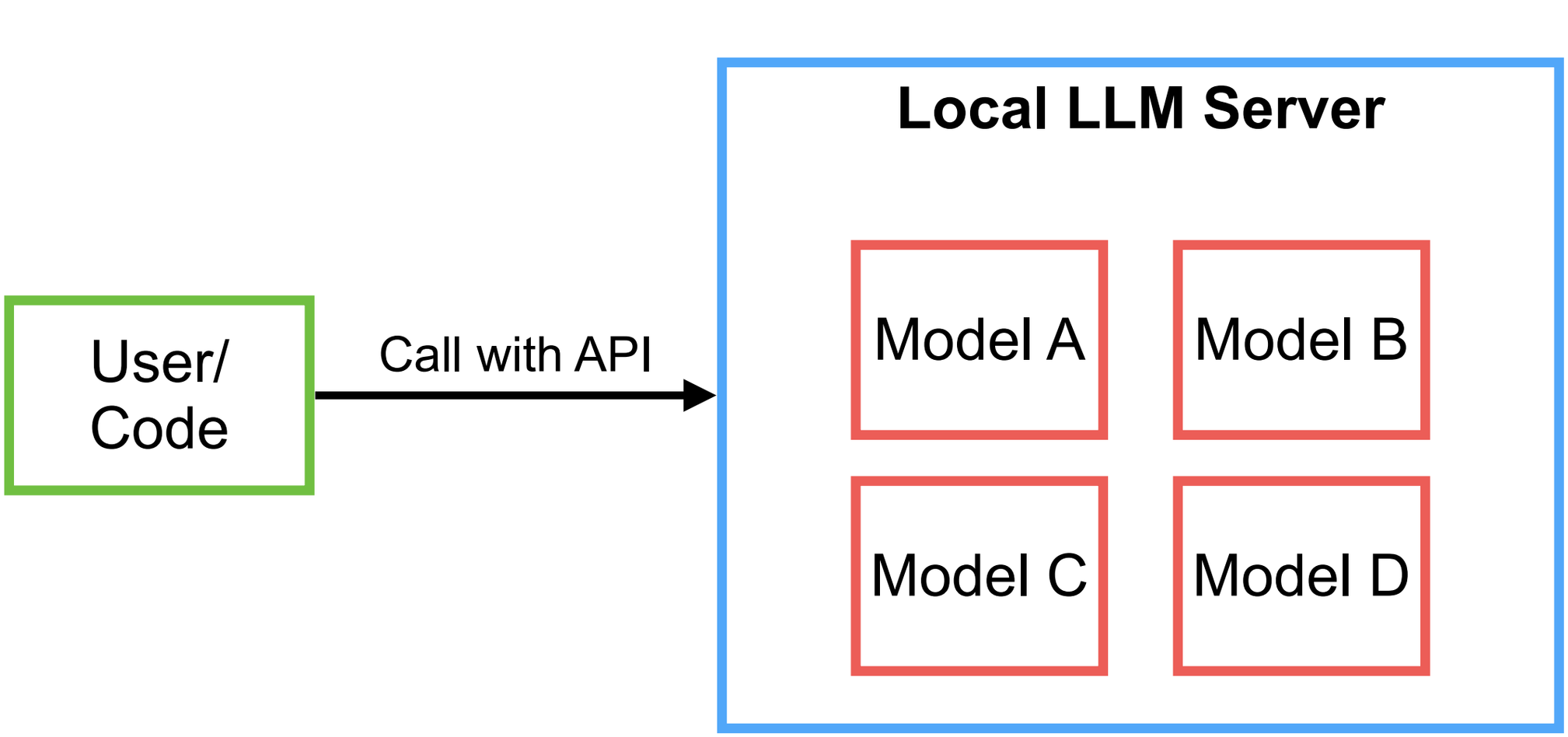
Local LLM

Run LLM on local machine/device
Try to customize with your requirement

- Reduce cost
- Data privacy
- Responsive
- Offline mode

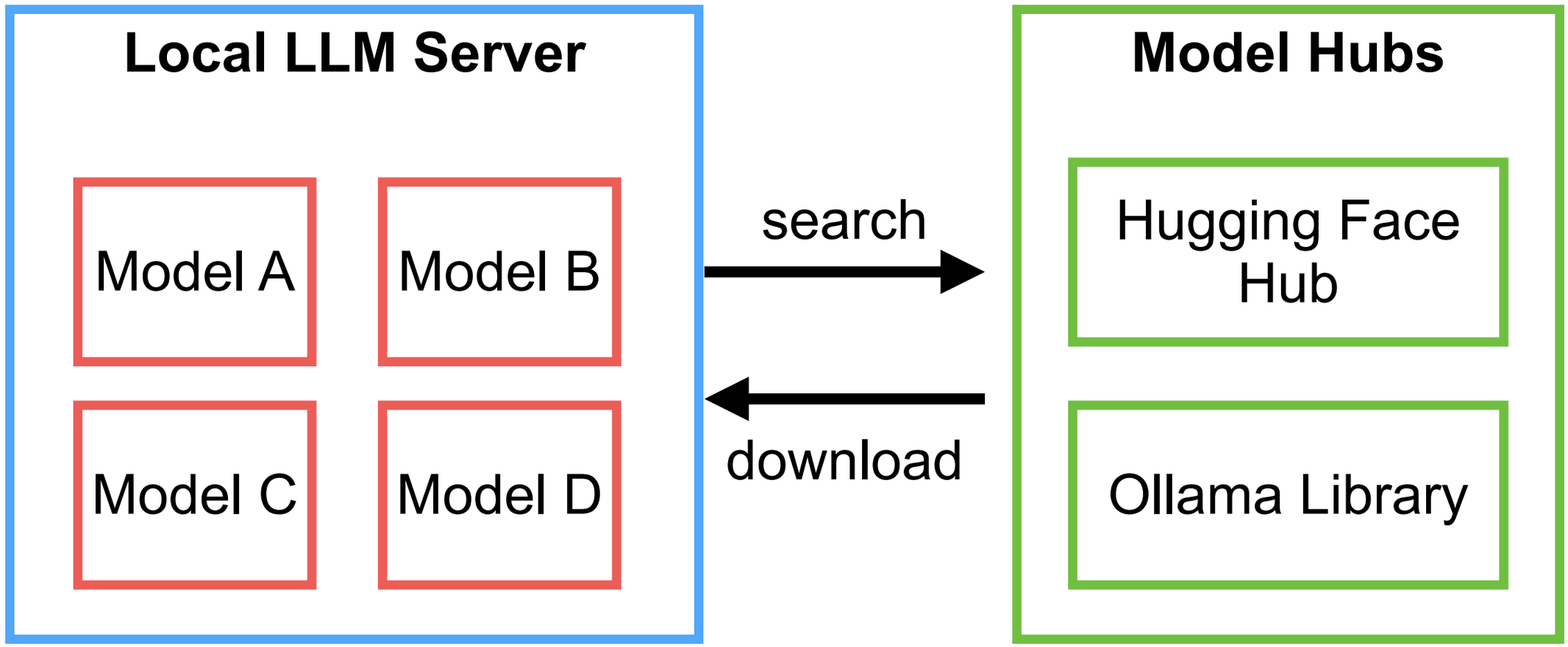
Local LLM

Improve your LLM models, more accurate answer



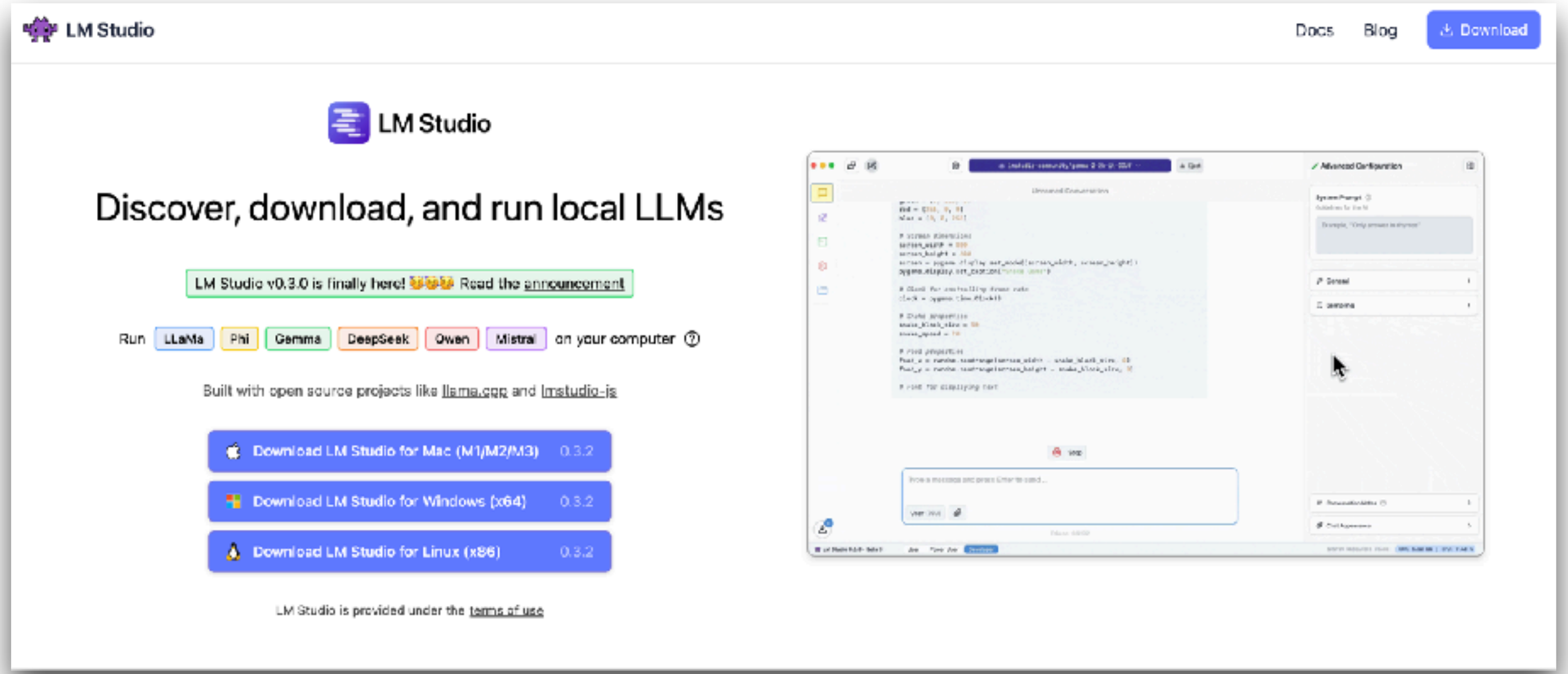
Models ?

How to download models ?



Local LLM Server

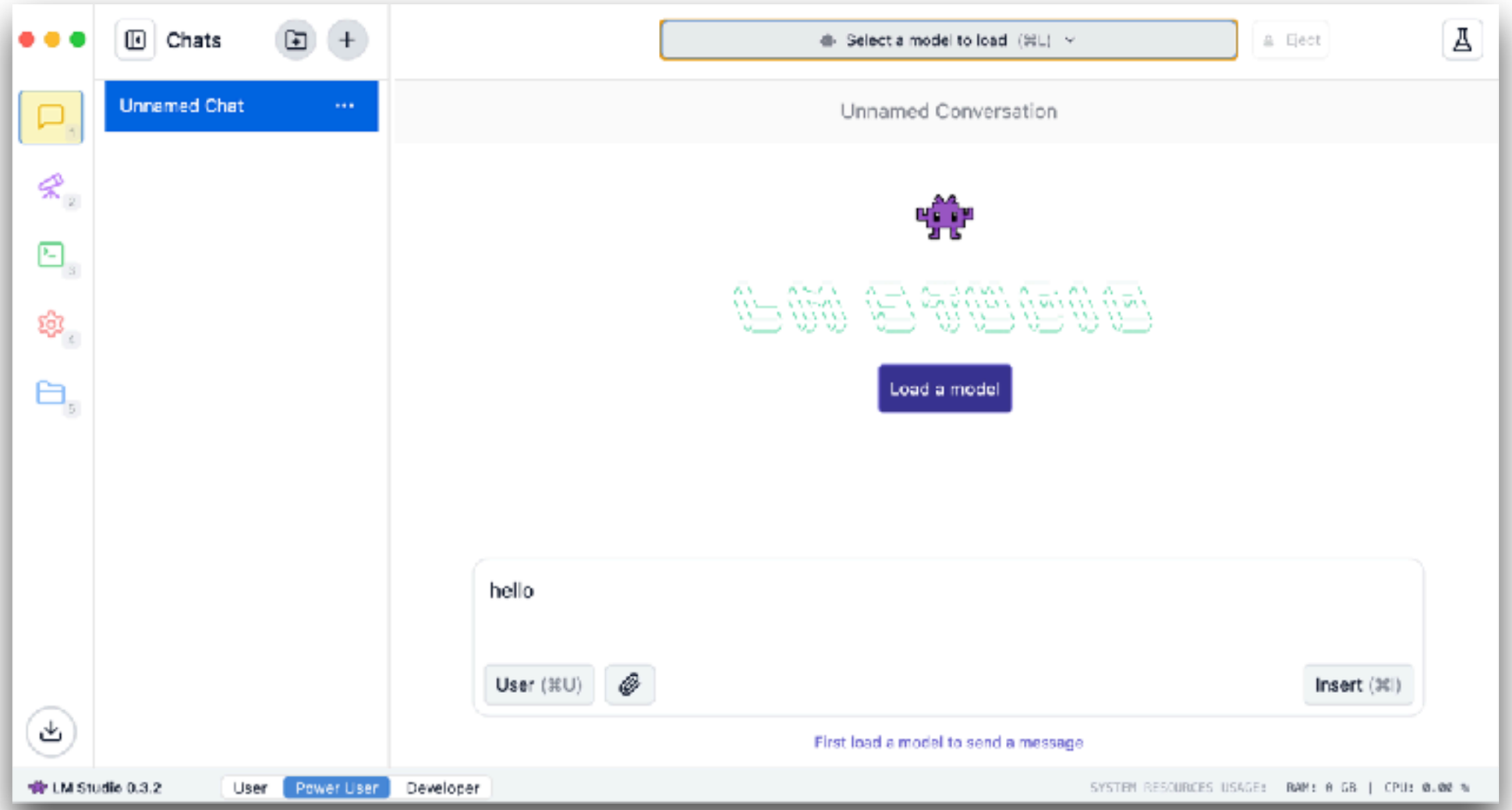
Local LLM with LM Studio



https://lmstudio.ai/

Local LLM with LM Studio

Load model from Hugging Face

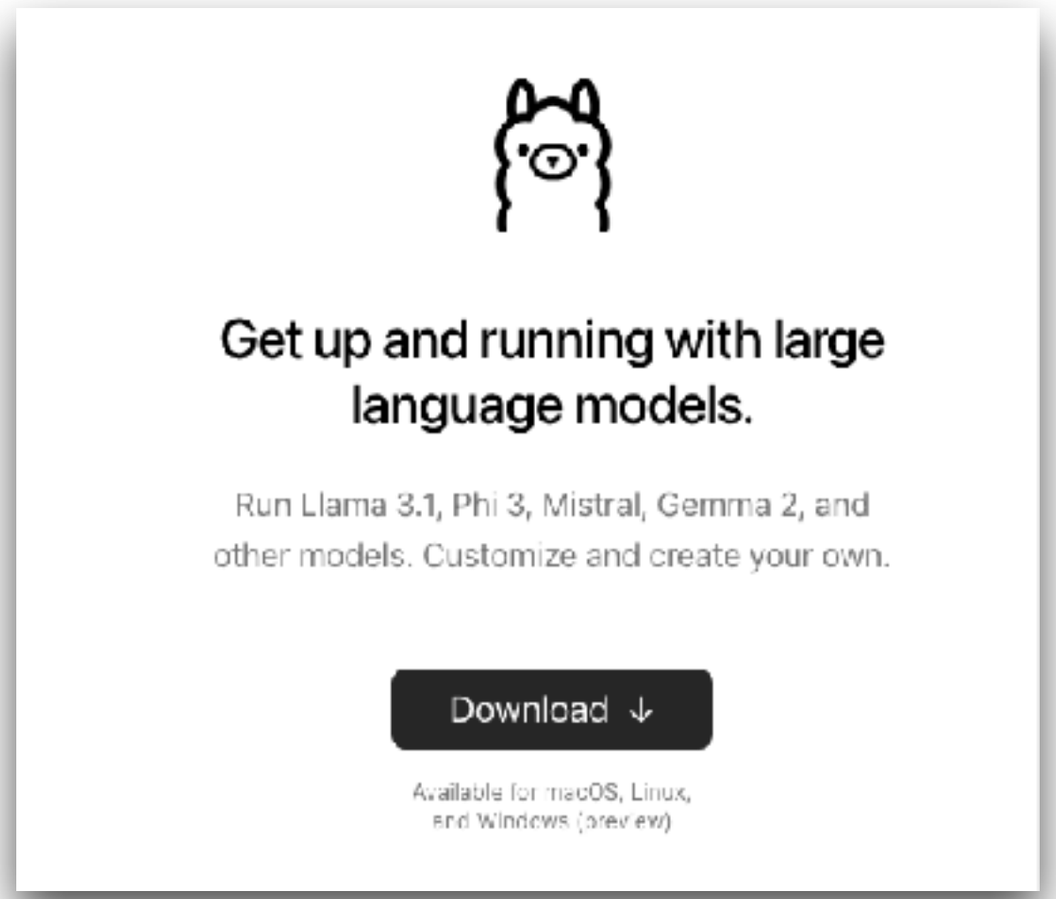


https://lmstudio.ai/

Local LLM with Ollama

$ollama run **llama3.1**

LLM model



https://ollama.com/

Local LLM with LocalAI

https://localai.io/

More

GPT4All

LlamaFile

Jan.ai

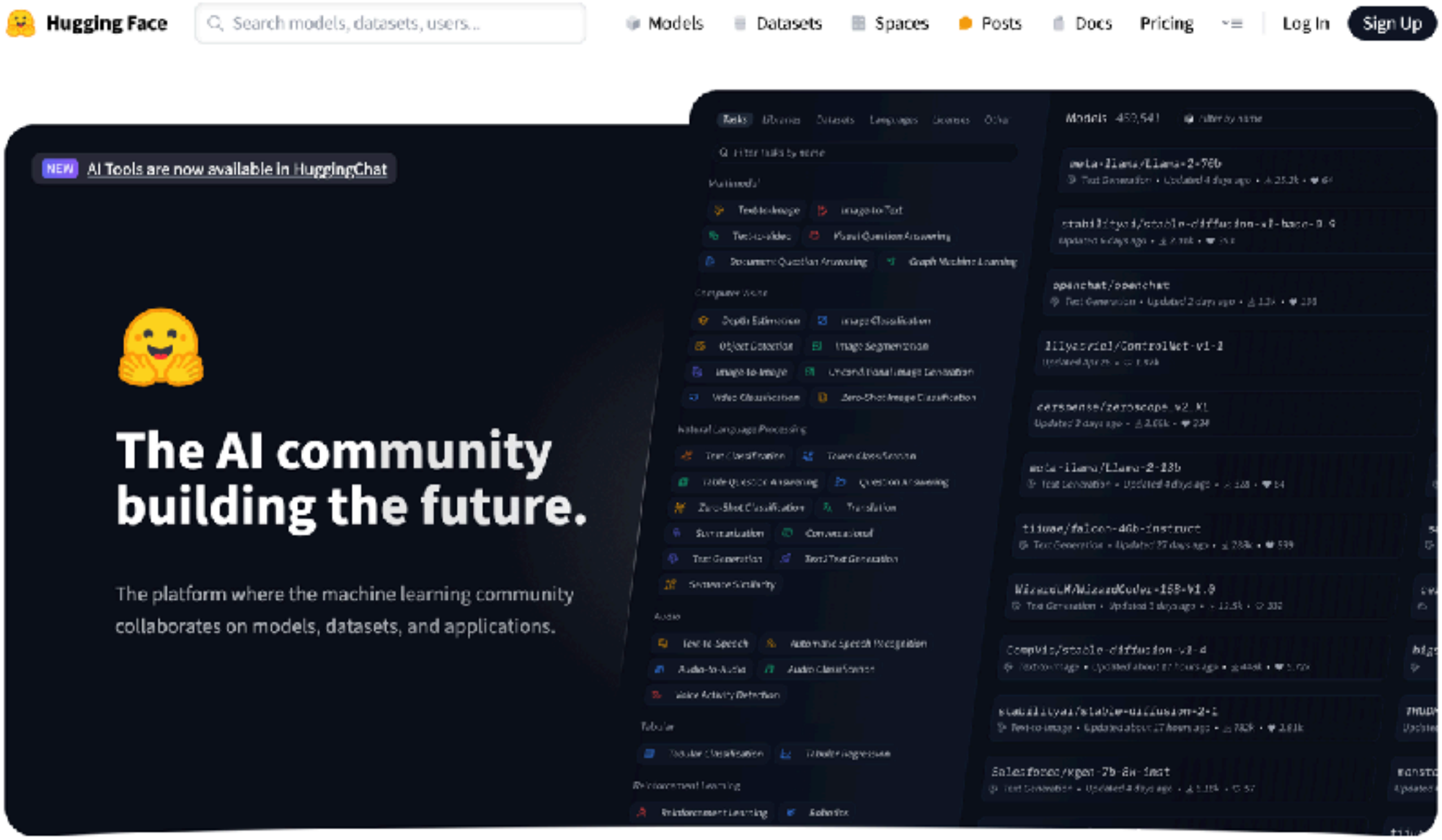
NextChat

Anything LLM

https://github.com/Hannibal046/Awesome-LLM

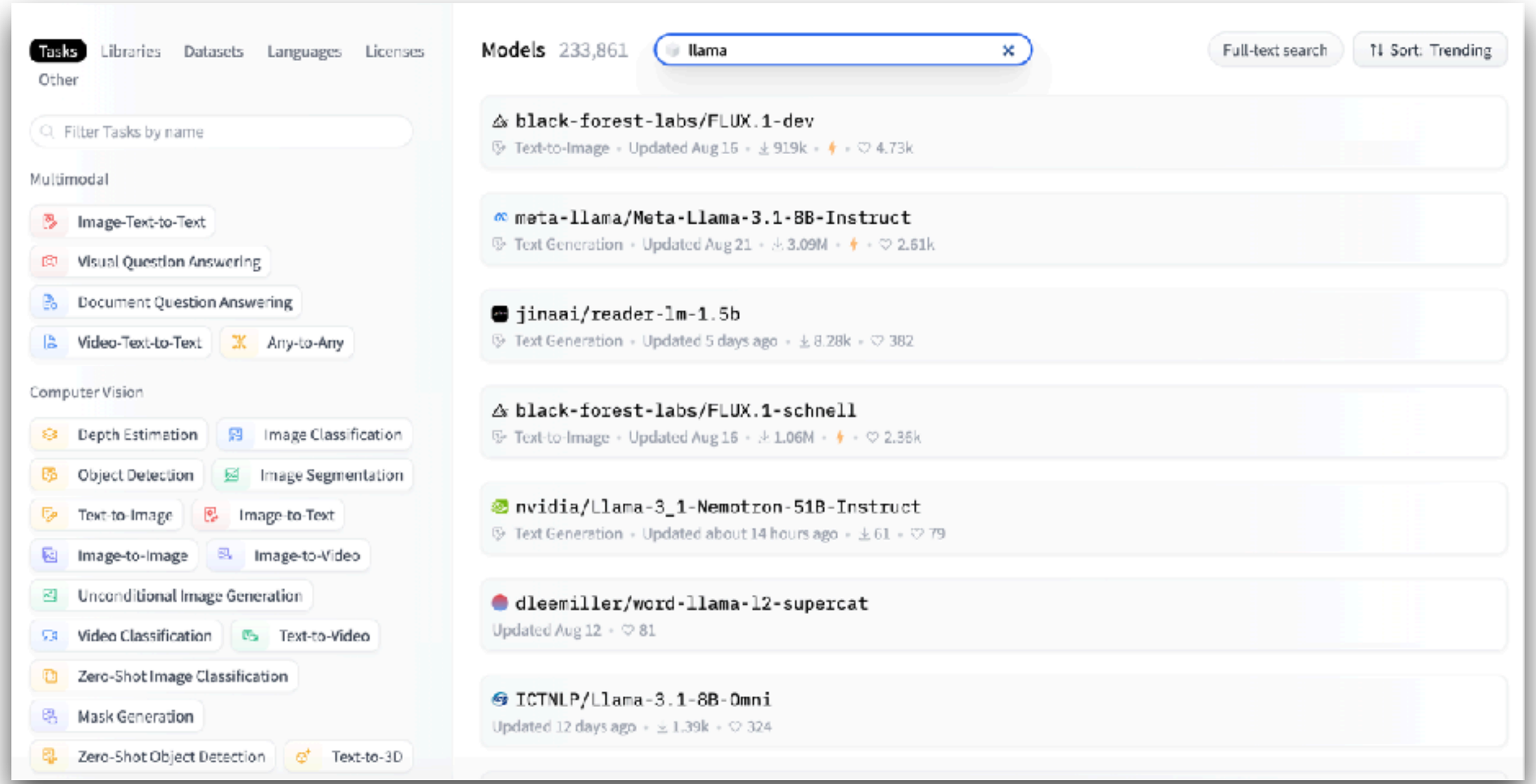
LLM Models

Hugging Face Model Hub



https://huggingface.co/

Hugging Face :: Model



https://huggingface.co/

Big Code model leader board

⭐ Big Code Models Leaderboard

Inspired from the 🤗 Open LLM Leaderboard and 🤗 Open LLM-Perf Leaderboard 🏋️, we compare performance of base multilingual code generation models on HumanEval benchmark and MultiPL-E. We also measure throughput and provide information about the models. We only compare open pre-trained multilingual code models, that people can start from as base models for their trainings.

🔍 Evaluation table | 📊 Performance Plot | 📝 About | Submit results 🚀

➡️ See All Columns ◀

🔍 Search for your model and press ENTER...

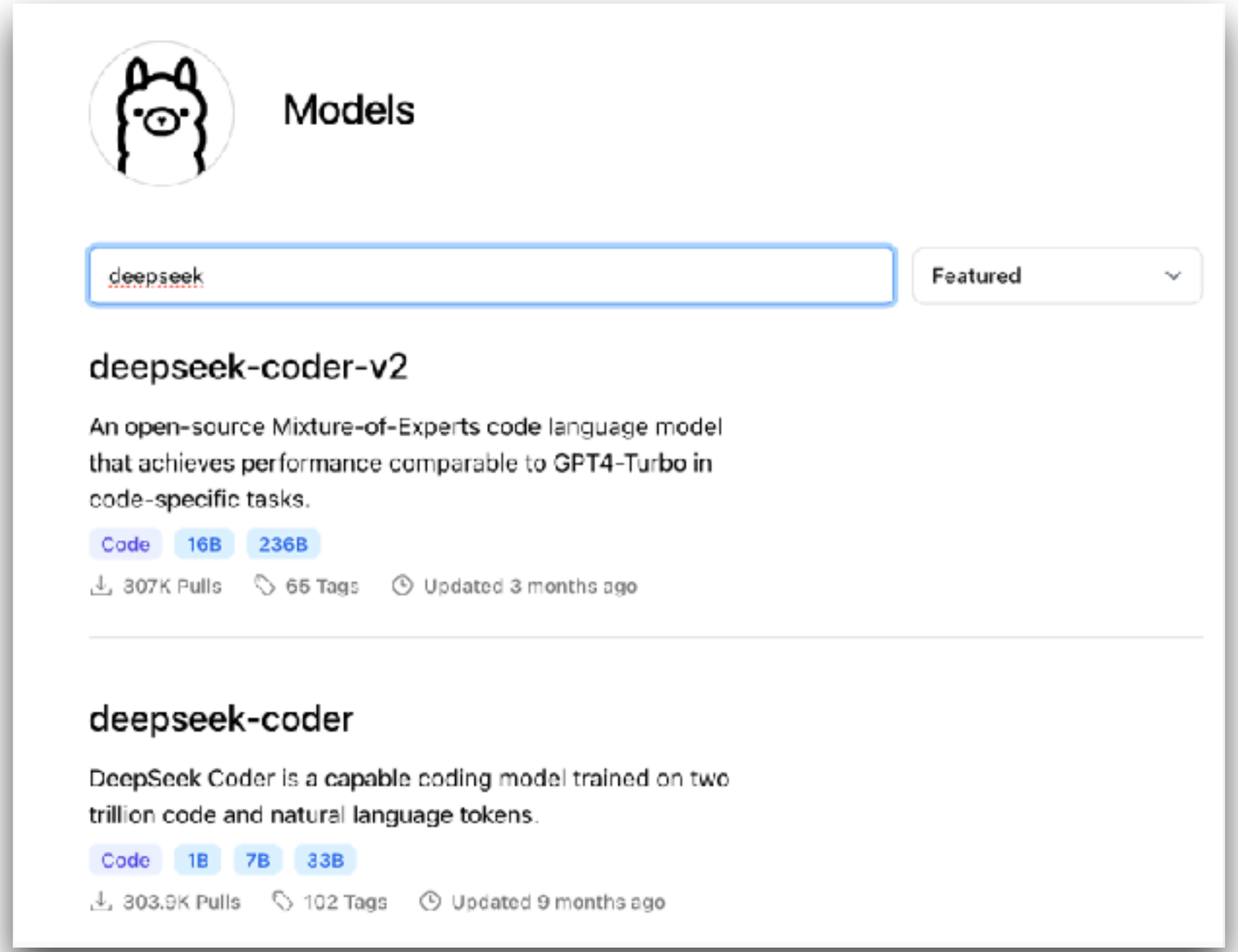
Filter model types

all | base | 🔶 instruction-tuned | EXT external-evaluation

T	Model	Win Rate	humaneval-python	java	javascript	cpp
🔶 EXT	OpenCodeInterpreter-DS-33B	55.83	75.23	54.8	69.06	64.47
🔶 EXT	Nxcode-CQ-7B-orpo	55.42	87.23	60.91	71.69	68.84
🔶	CodeQwen1.5-7B-Chat	55.08	87.2	61.04	70.31	67.85
🔶 EXT	CodeFuse-DeepSeek-33b	54.33	76.83	60.76	66.46	65.22
🔶 EXT	DeepSeek-Coder-33b-instruct	52	80.02	52.03	65.13	62.36
🔶 EXT	Artigenz-Coder-DS-6.7B	51.5	70.89	56.84	66.16	59.75
🔶 EXT	DeepSeek-Coder-7b-instruct	50.33	80.22	53.34	66.8	59.66
🔶 EXT	OpenCodeInterpreter-DS-6.7B	49.67	73.2	51.41	63.85	60.81

https://huggingface.co/spaces/bigcode/bigcode-models-leaderboard

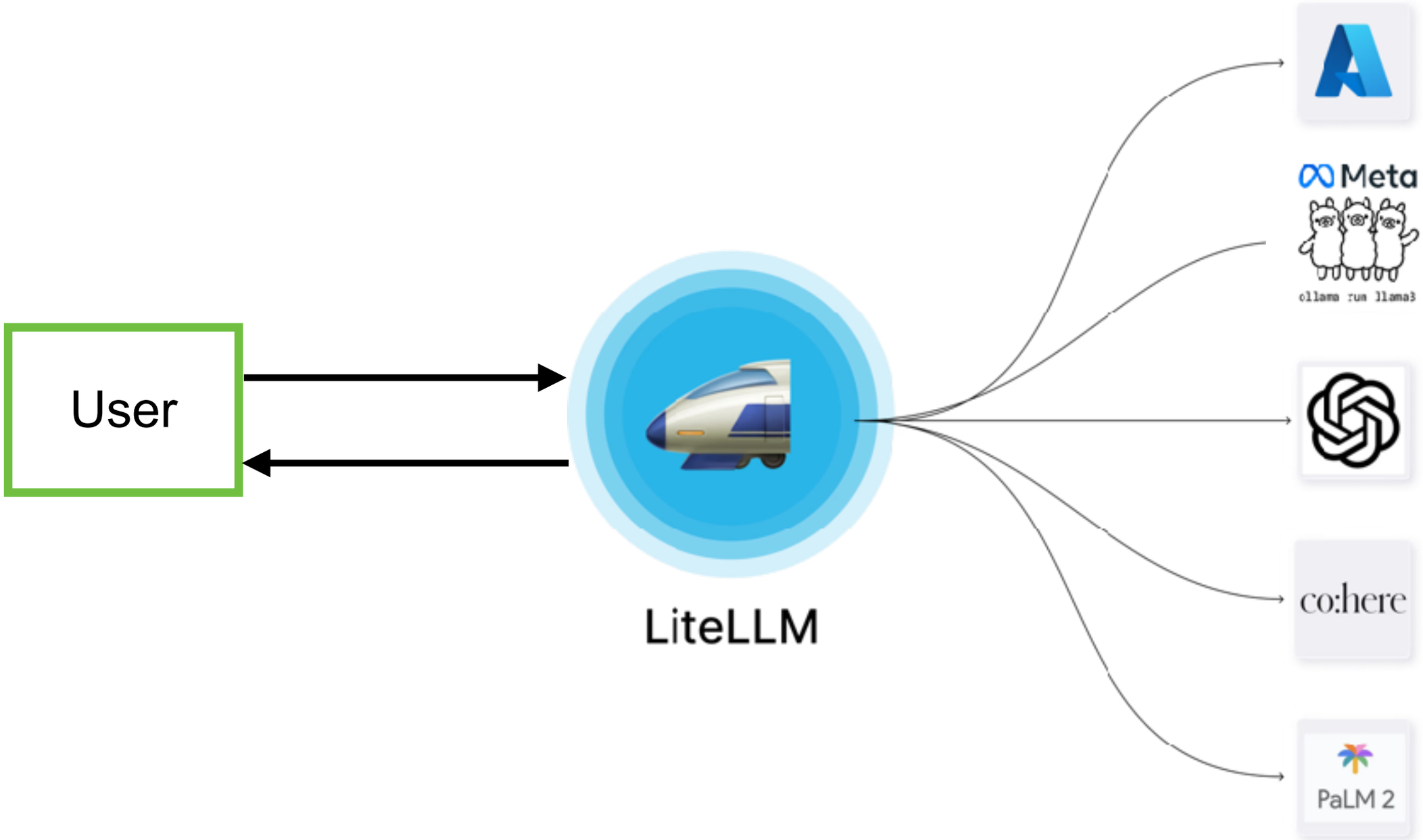
Model in Ollama



https://ollama.com/library

LiteLLM as a Proxy

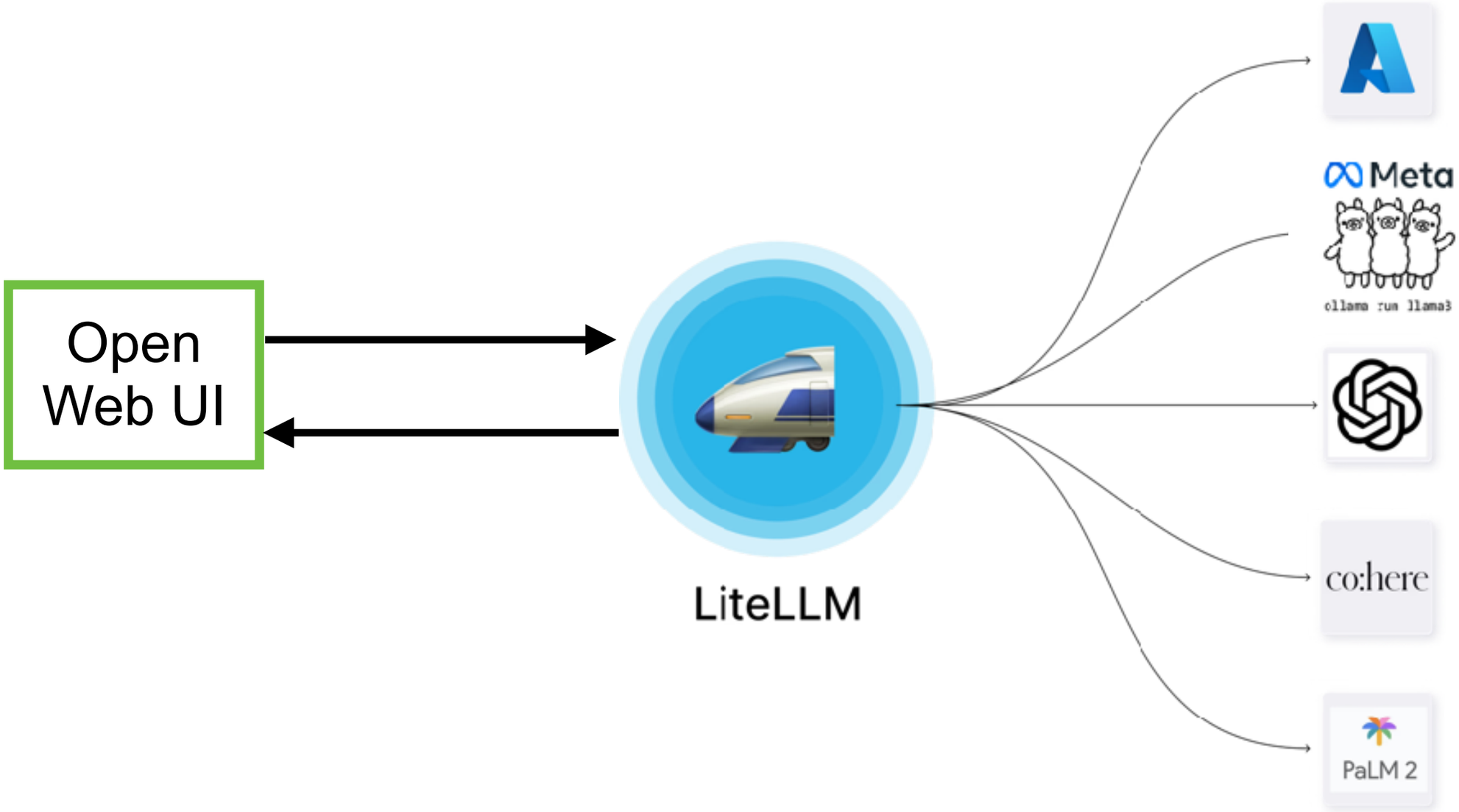
LiteLLM as a Proxy



https://www.litellm.ai/

Workshop

Use docker compose to build and run



https://docs.openwebui.com/

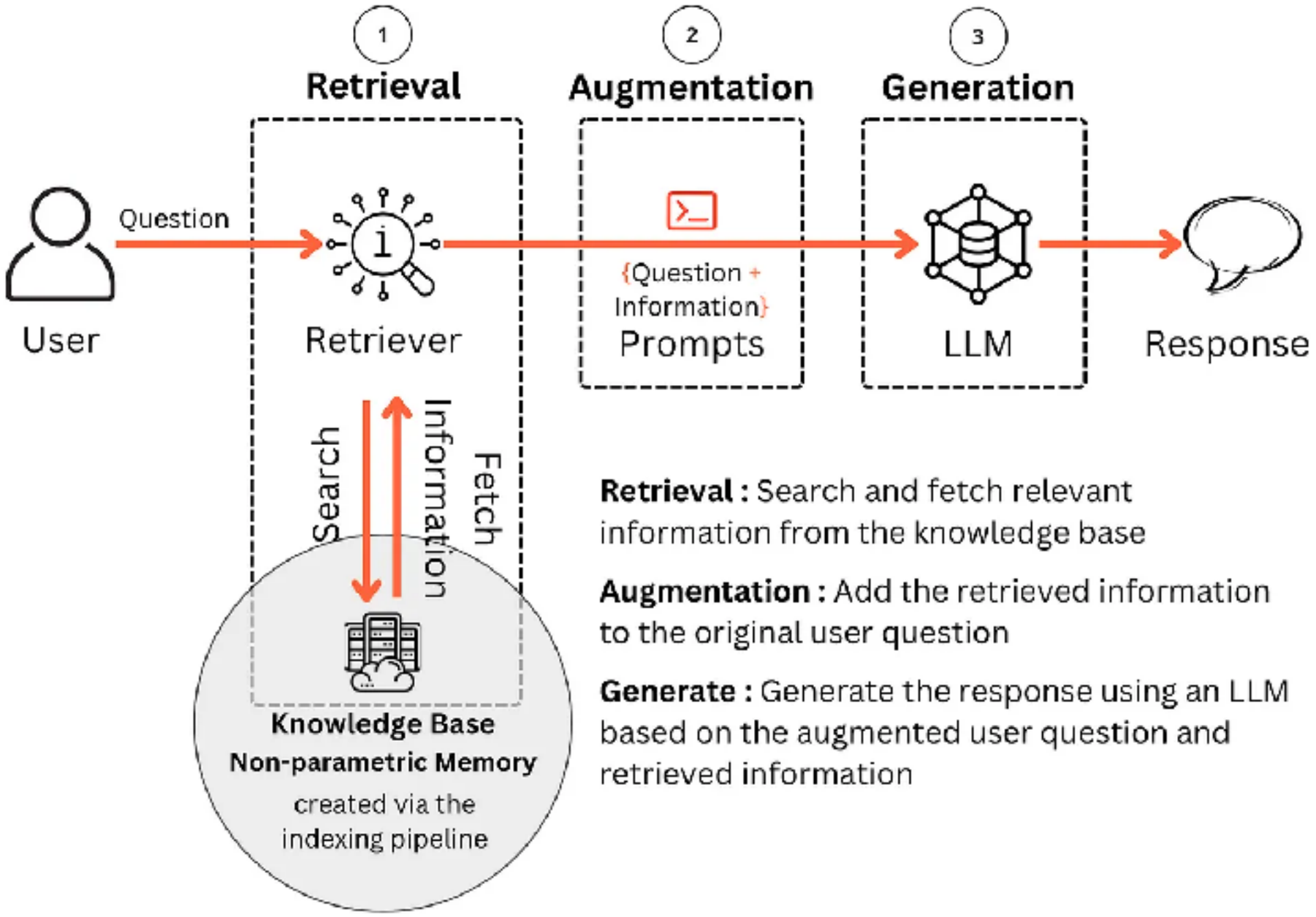
Retrieval-Augmented Generation (RAG)

What is RAG ?

Enhance LLM with external knowledge
Improve your LLM models, more accurate answer

Proprietary knowledge	Up-to-data Information
Citing sources	Data security Access control List (ACL)

RAG



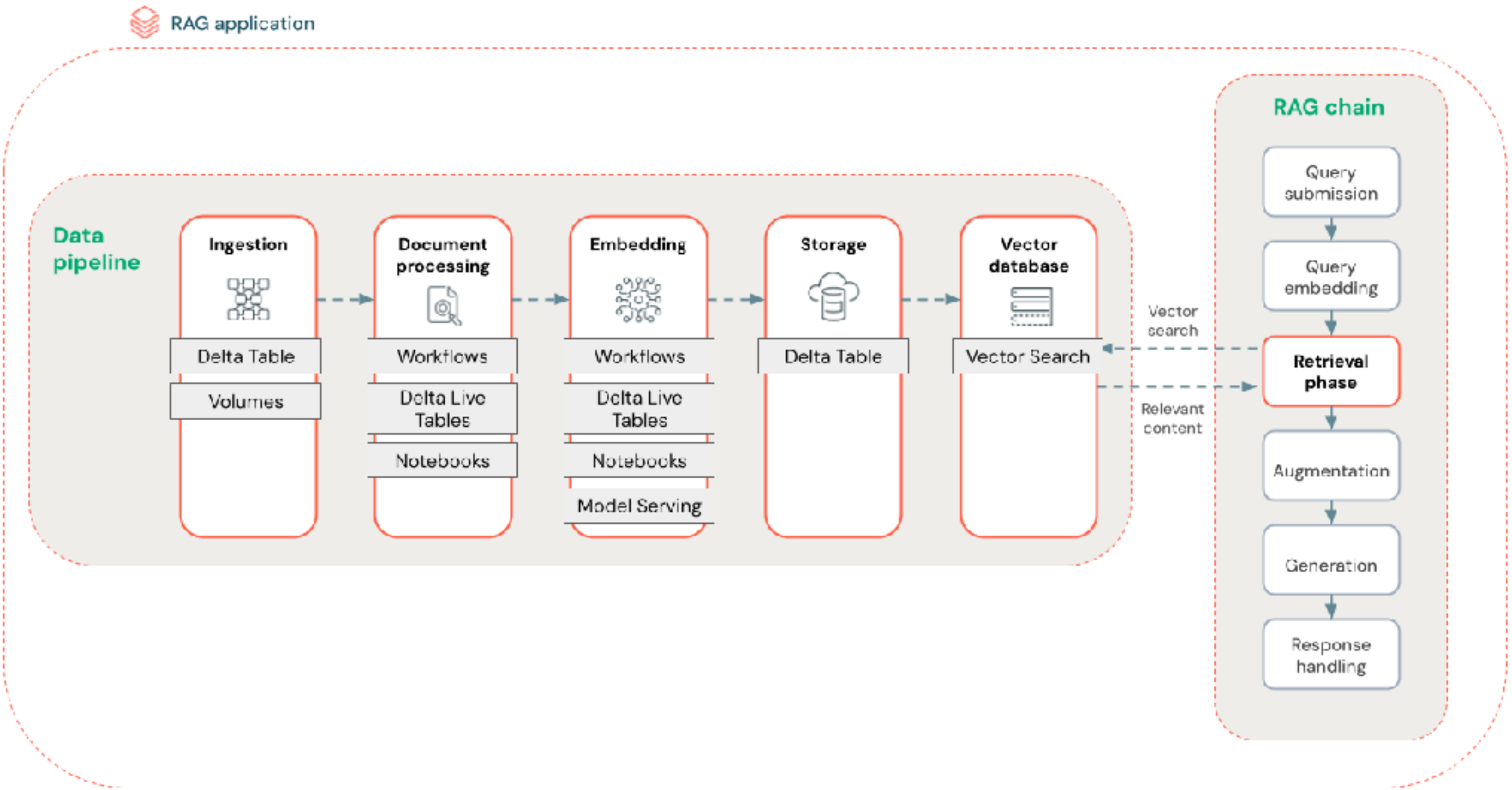
Retrieval : Search and fetch relevant information from the knowledge base

Augmentation : Add the retrieved information to the original user question

Generate : Generate the response using an LLM based on the augmented user question and retrieved information

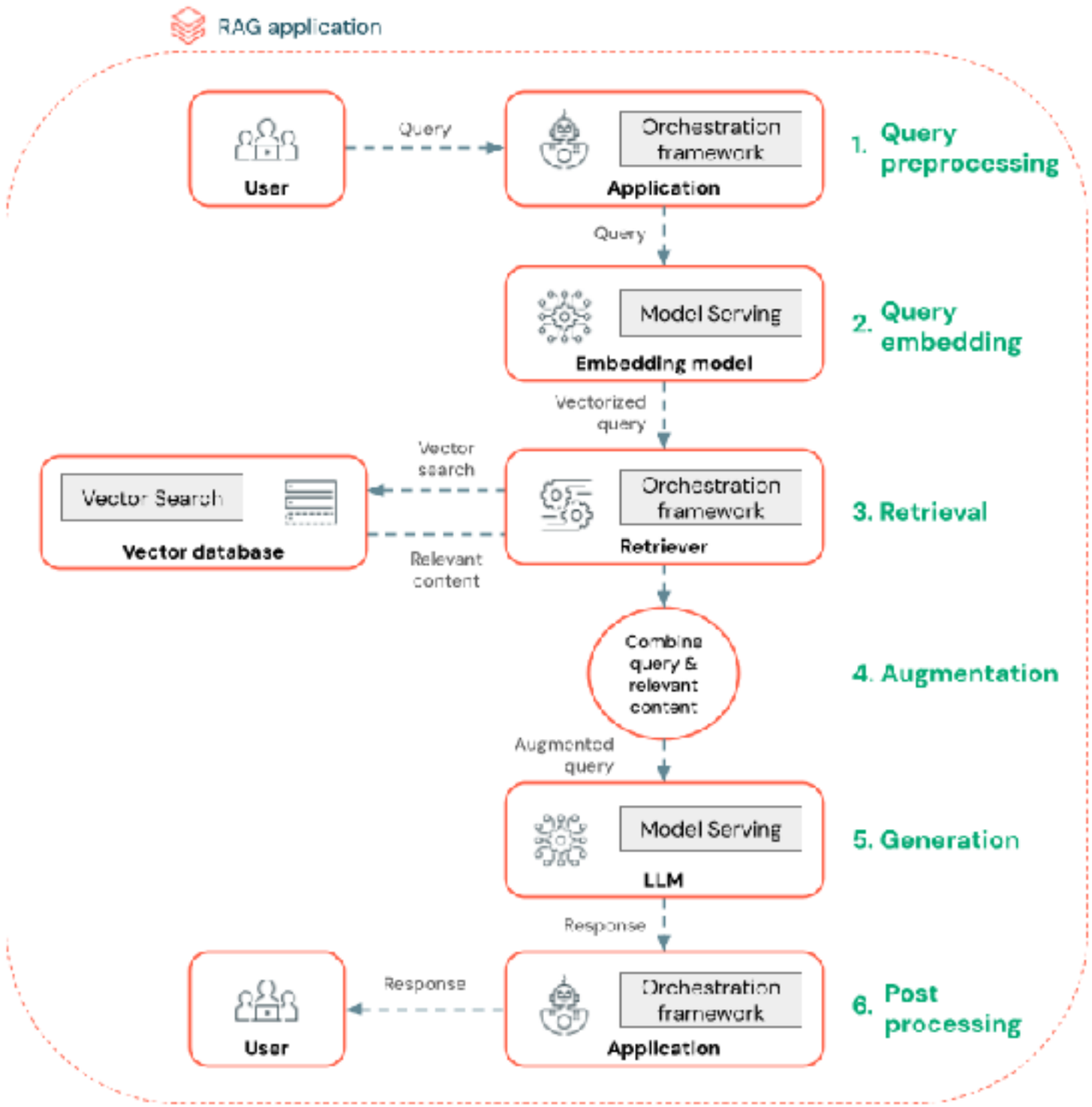
https://docs.databricks.com/en/generative-ai/retrieval-augmented-generation.html

RAG Data Pipeline



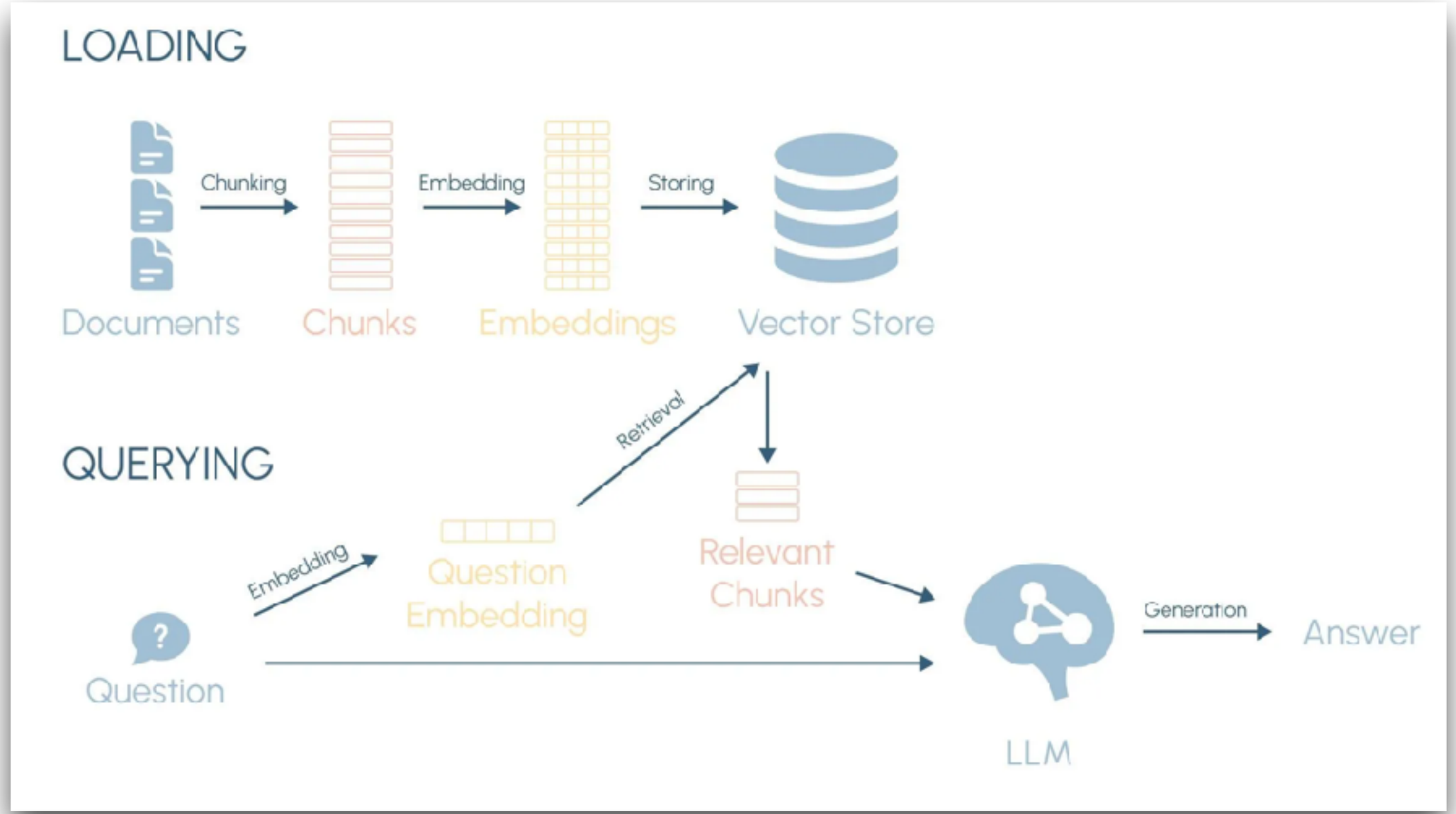
https://docs.databricks.com/en/generative-ai/retrieval-augmented-generation.html

RAG Agent



https://docs.databricks.com/en/generative-ai/retrieval-augmented-generation.html

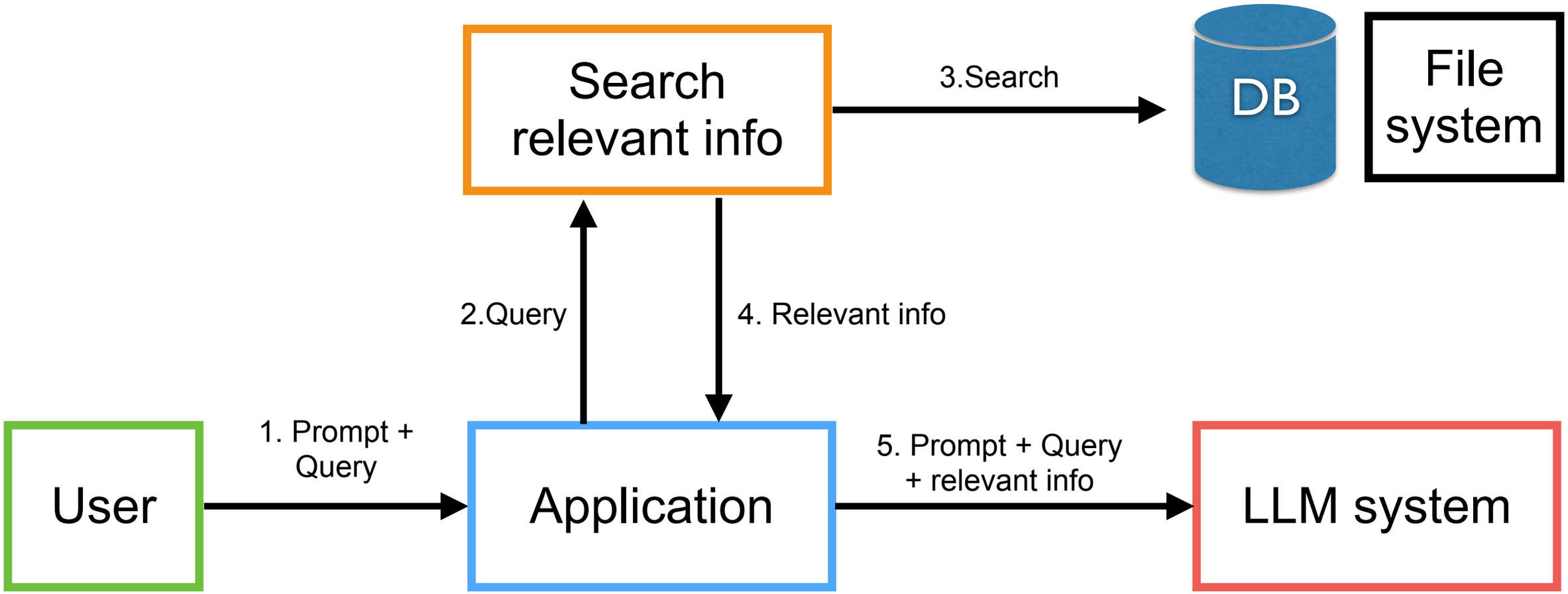
RAG Process



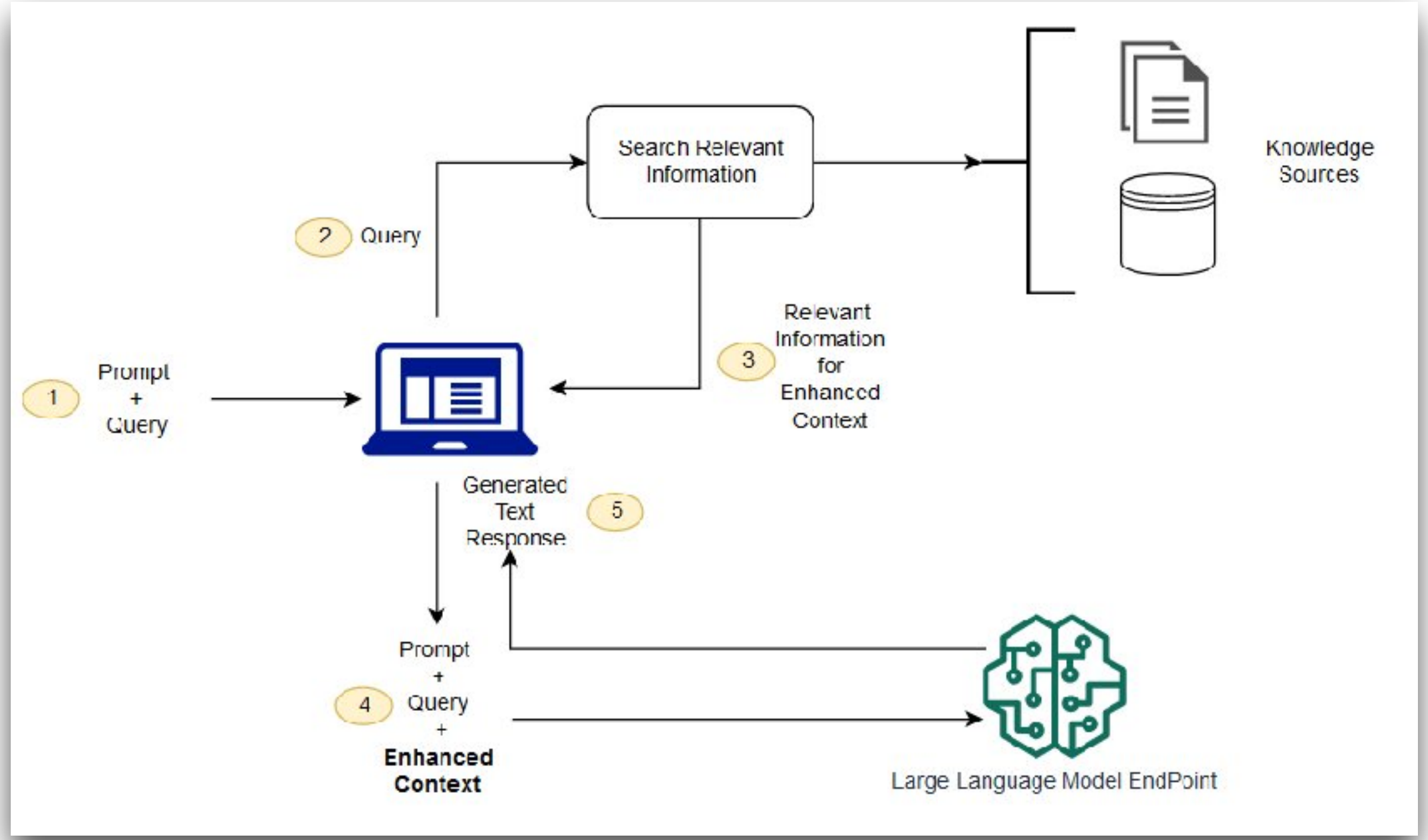
https://medium.com/@codeawake/ai-chatbot-5bd2fa3324e3

RAG with LLM

Improve your LLM models, more accurate answer

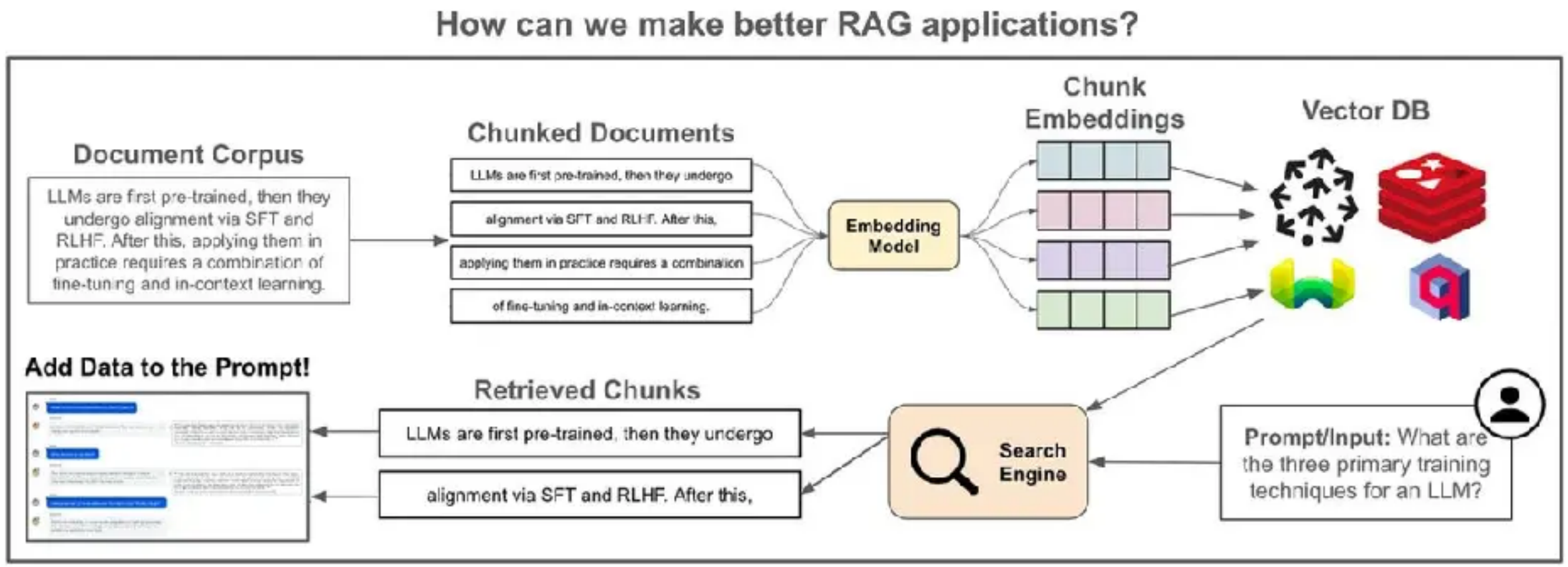


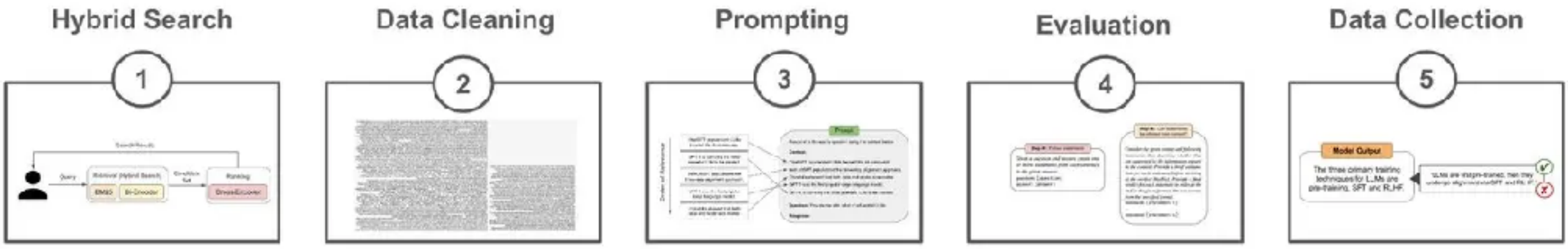
RAG with LLM



https://aws.amazon.com/th/what-is/retrieval-augmented-generation/

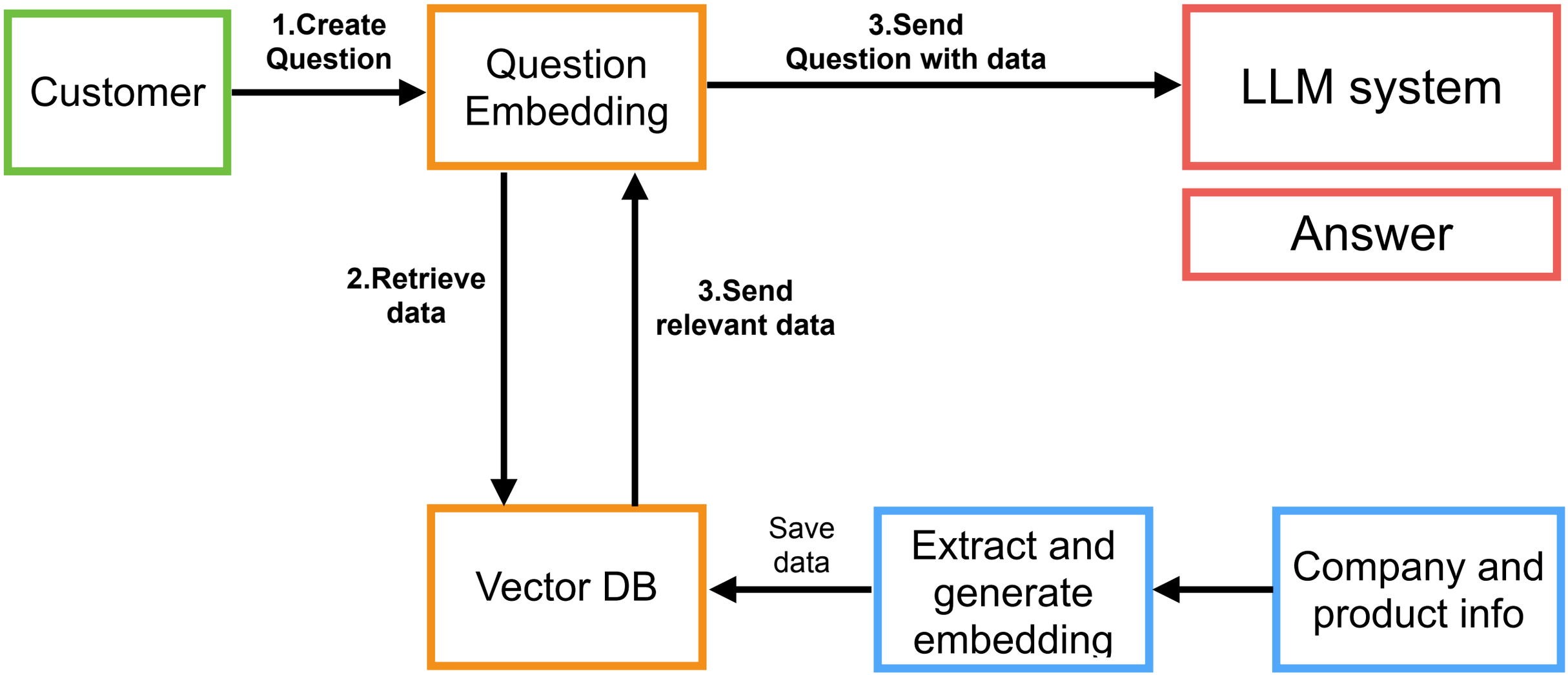
Better RAG application





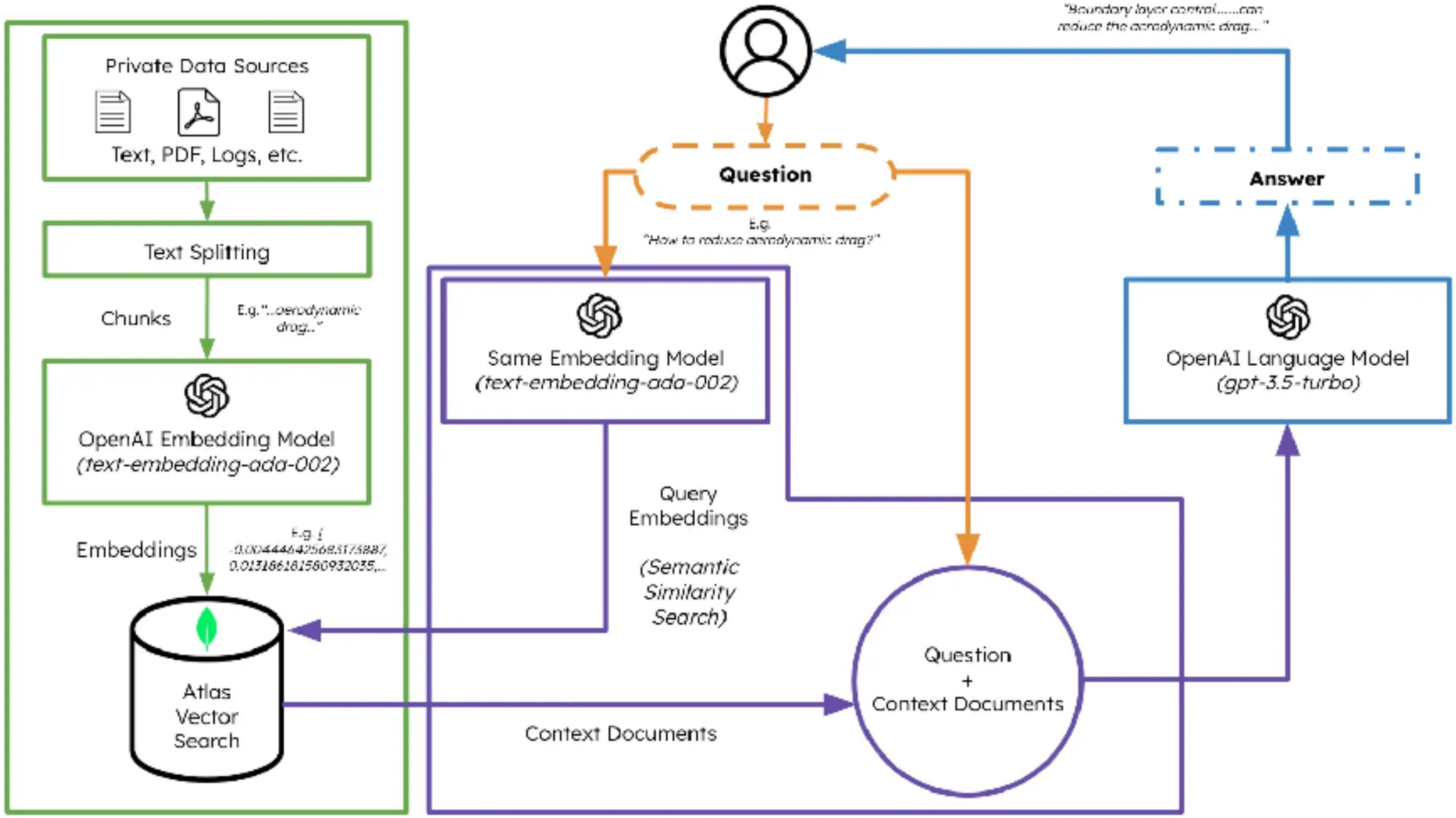
https://stackoverflow.blog/2024/08/15/practical-tips-for-retrieval-augmented-generation-rag/

Demo with Chatbot



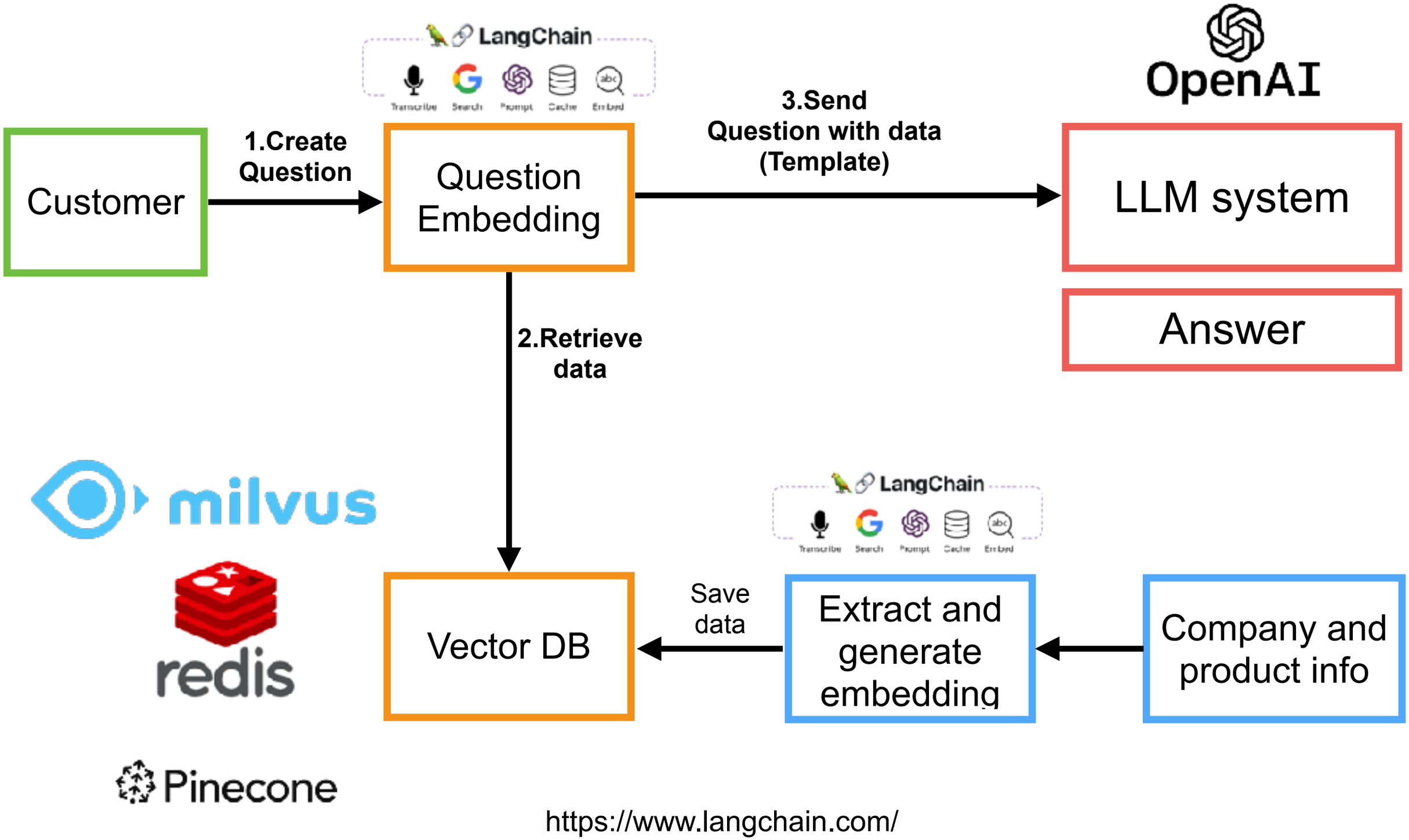
https://redis.io/blog/build-ecommerce-chatbot-with-redis/

RAG with MongoDB



https://www.mongodb.com/developer/products/atlas/taking-rag-to-production-documentation-ai-chatbot/

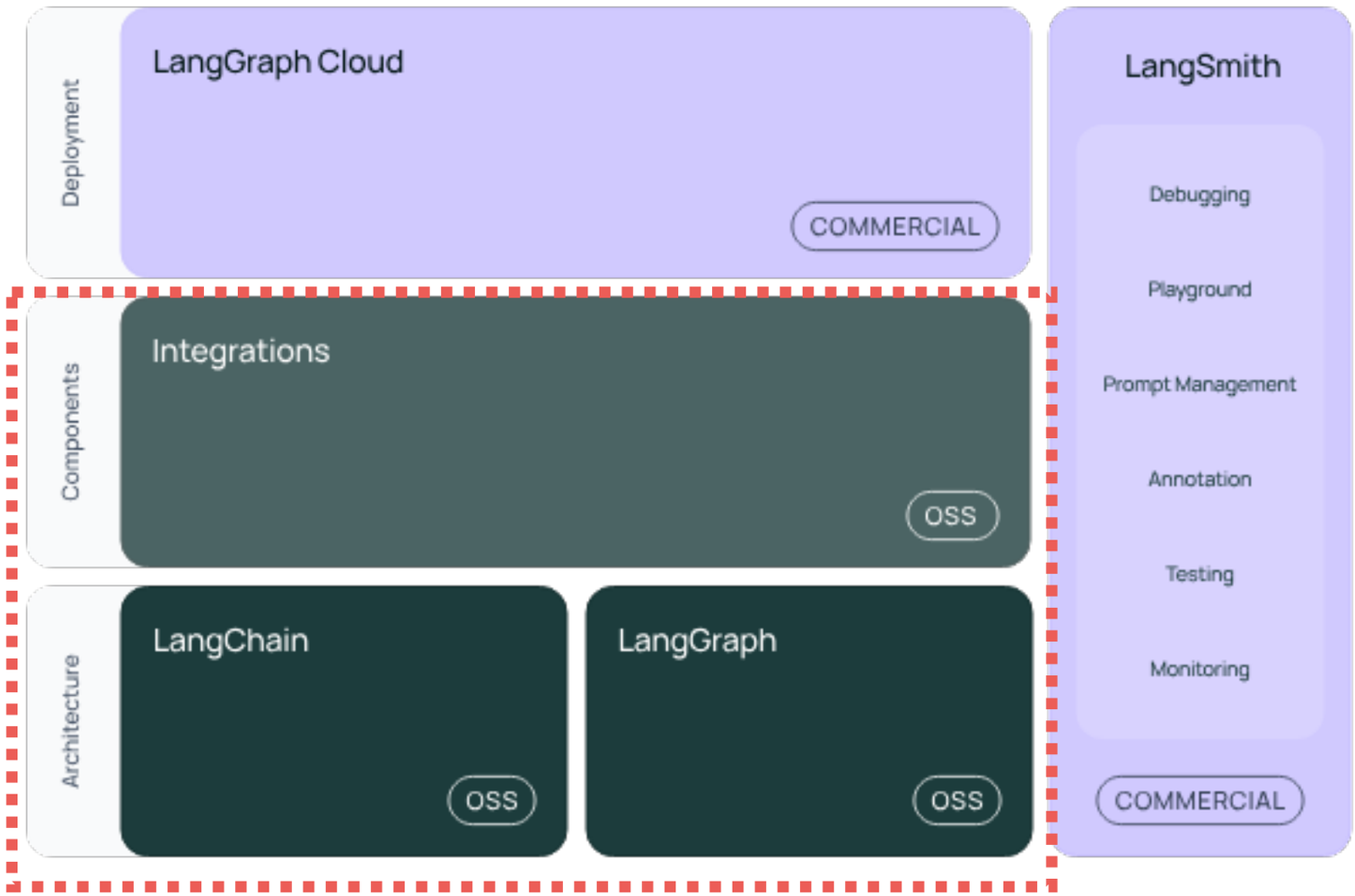
Frameworks and Tools



https://www.langchain.com/

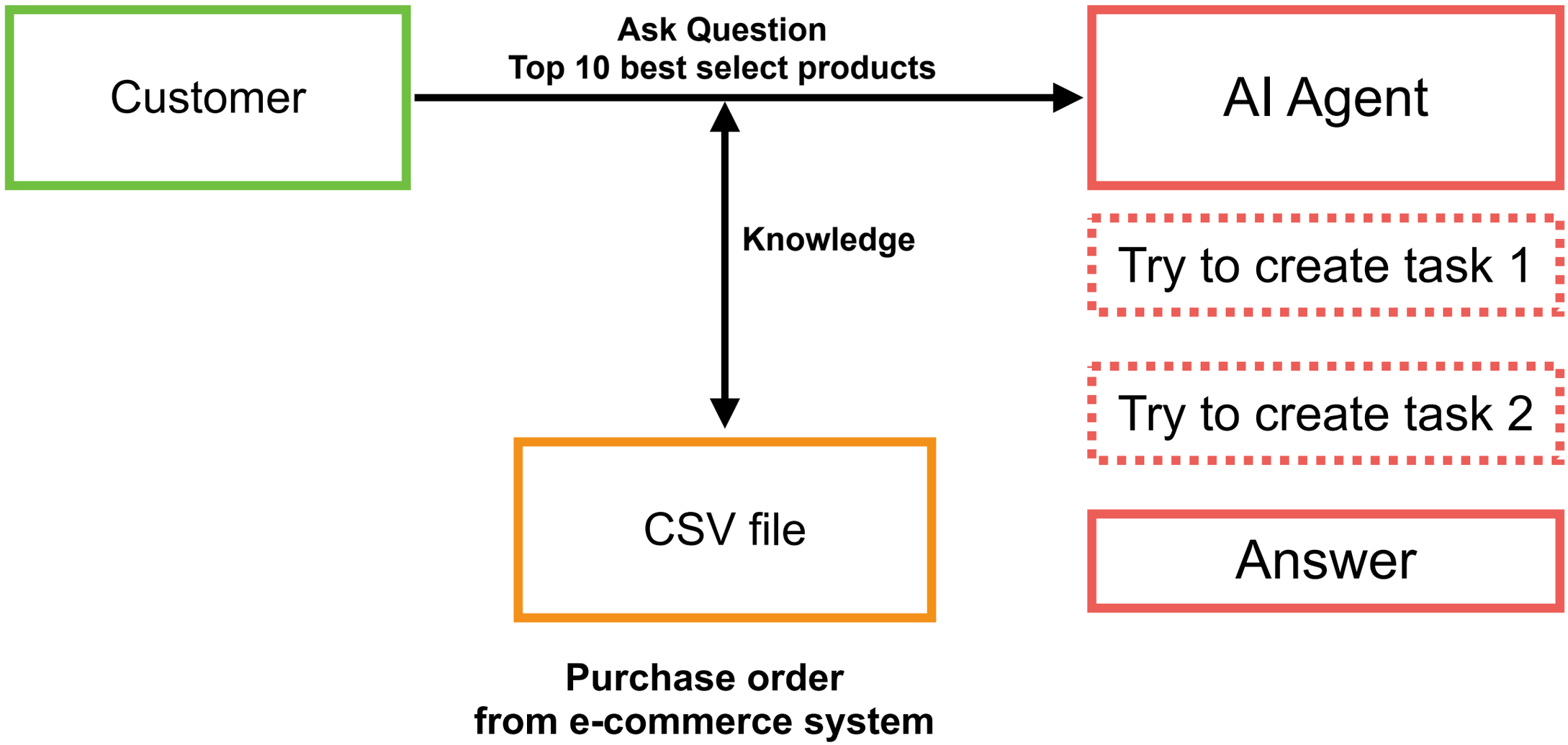
LangChain

Framework for develop application powered by LLM

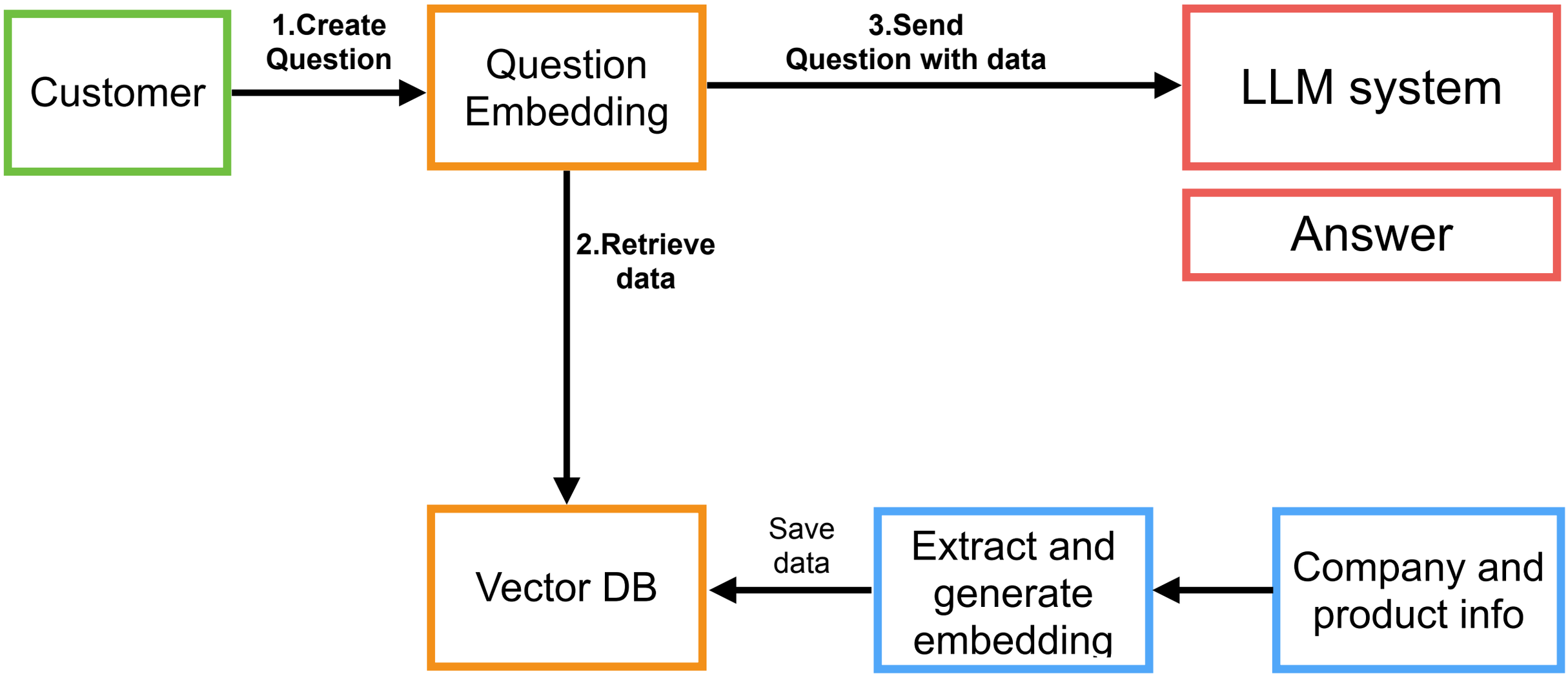


https://python.langchain.com/

Data Analysis Agent



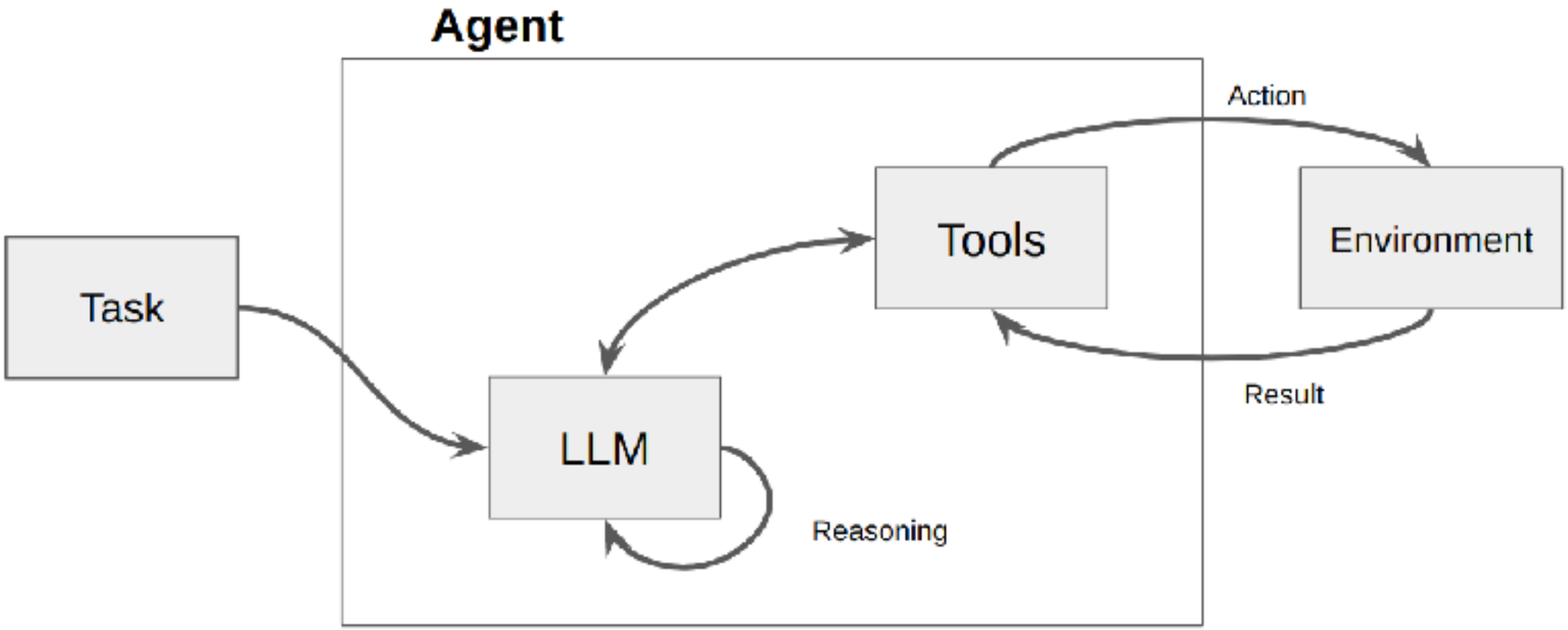
Steps to develop



https://redis.io/blog/build-ecommerce-chatbot-with-redis/

AI Agent

AI Agent



Workshop AI Agent with Aider

GPT-4o

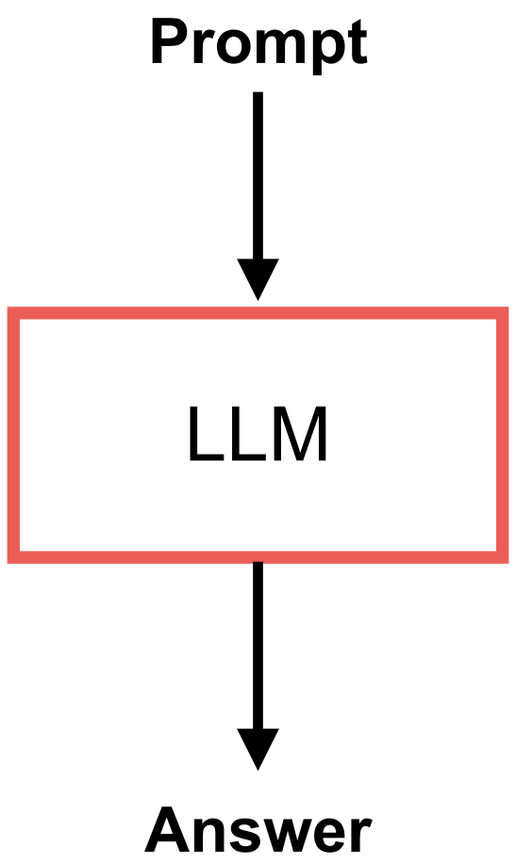
Claude 3.5 Sonnet

DeepSeek Coder

Ollama

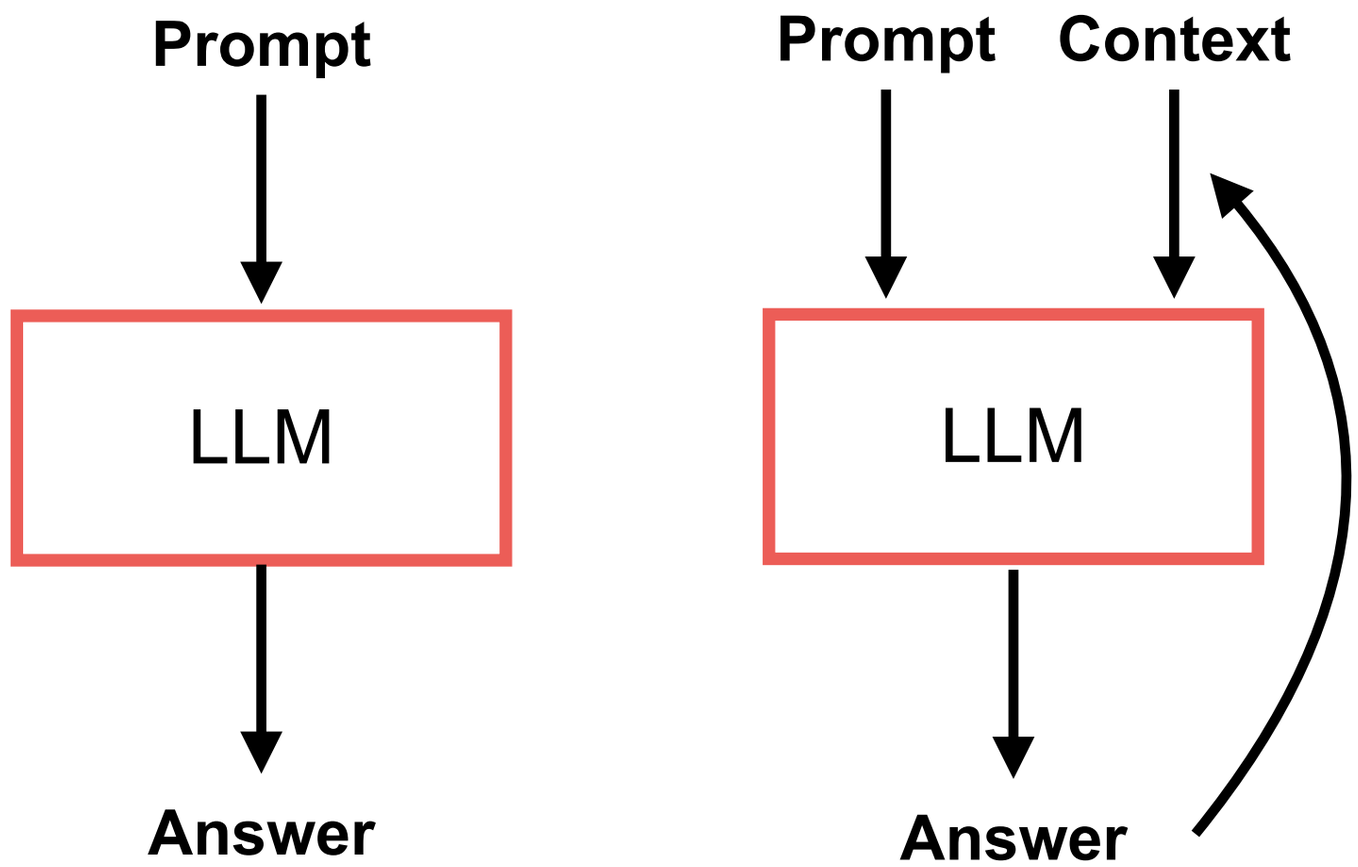
https://aider.chat/docs/llms/ollama.html

LLM



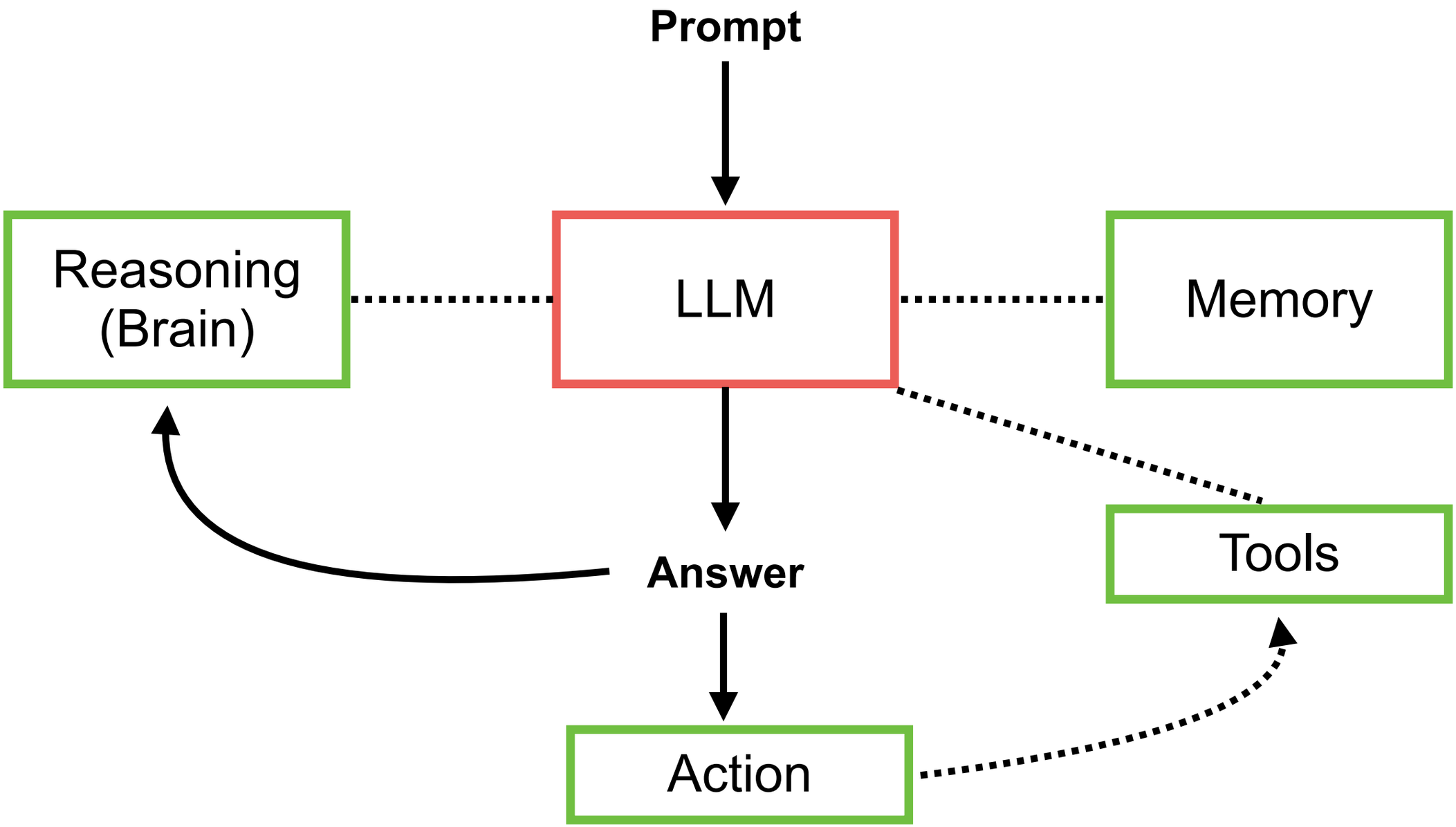
https://towardsdatascience.com/intro-to-llm-agents-with-langchain-when-rag-is-not-enough-7d8c08145834

LLM -> RAG



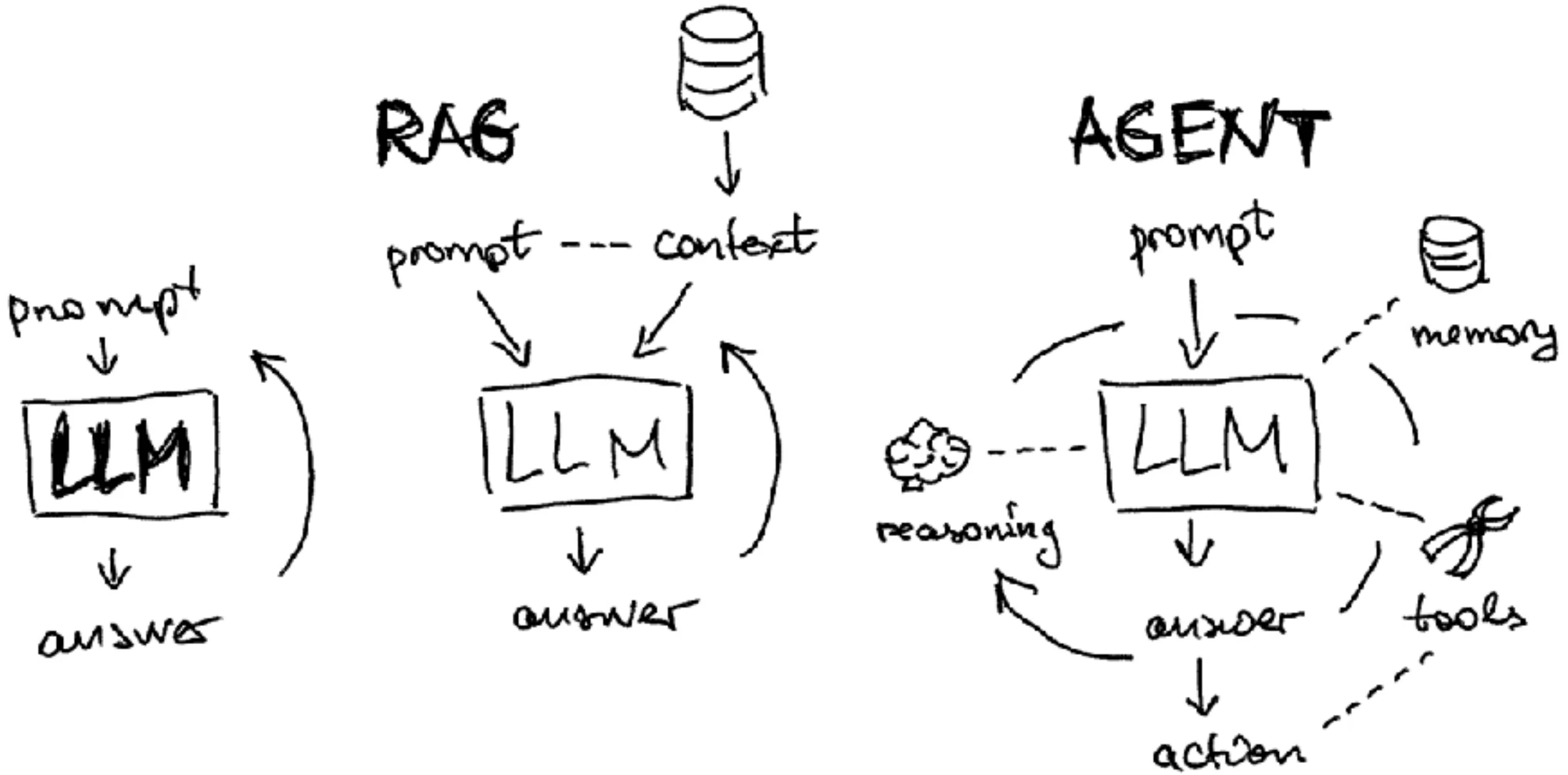
https://towardsdatascience.com/intro-to-llm-agents-with-langchain-when-rag-is-not-enough-7d8c08145834

Agent



https://towardsdatascience.com/intro-to-llm-agents-with-langchain-when-rag-is-not-enough-7d8c08145834

LLM -> RAG -> Agent



https://towardsdatascience.com/intro-to-llm-agents-with-langchain-when-rag-is-not-enough-7d8c08145834

Workshop Record Expense

Here is the extracted data from the payment slip in JSON format:


```
{
  "payment_status": "Completed",
  "date_time": "20 Aug 24 1:33 PM",
  "payer_name": "Mr. Somkiat P",
  "bank": "KBank",
  "payer_account": "xxx-x-x9323-x",
  "payee_name": "AMZ_SD4330 UNION MALL",
  "company_name": "DEEMAPHARUAY CO., LTD.",
  "transaction_id": "202408202155006",
  "payment_reference_id": "014233133358BQRO9410",
  "amount": "45.00 Baht",
  "fee": "0.00 Baht"
}
```

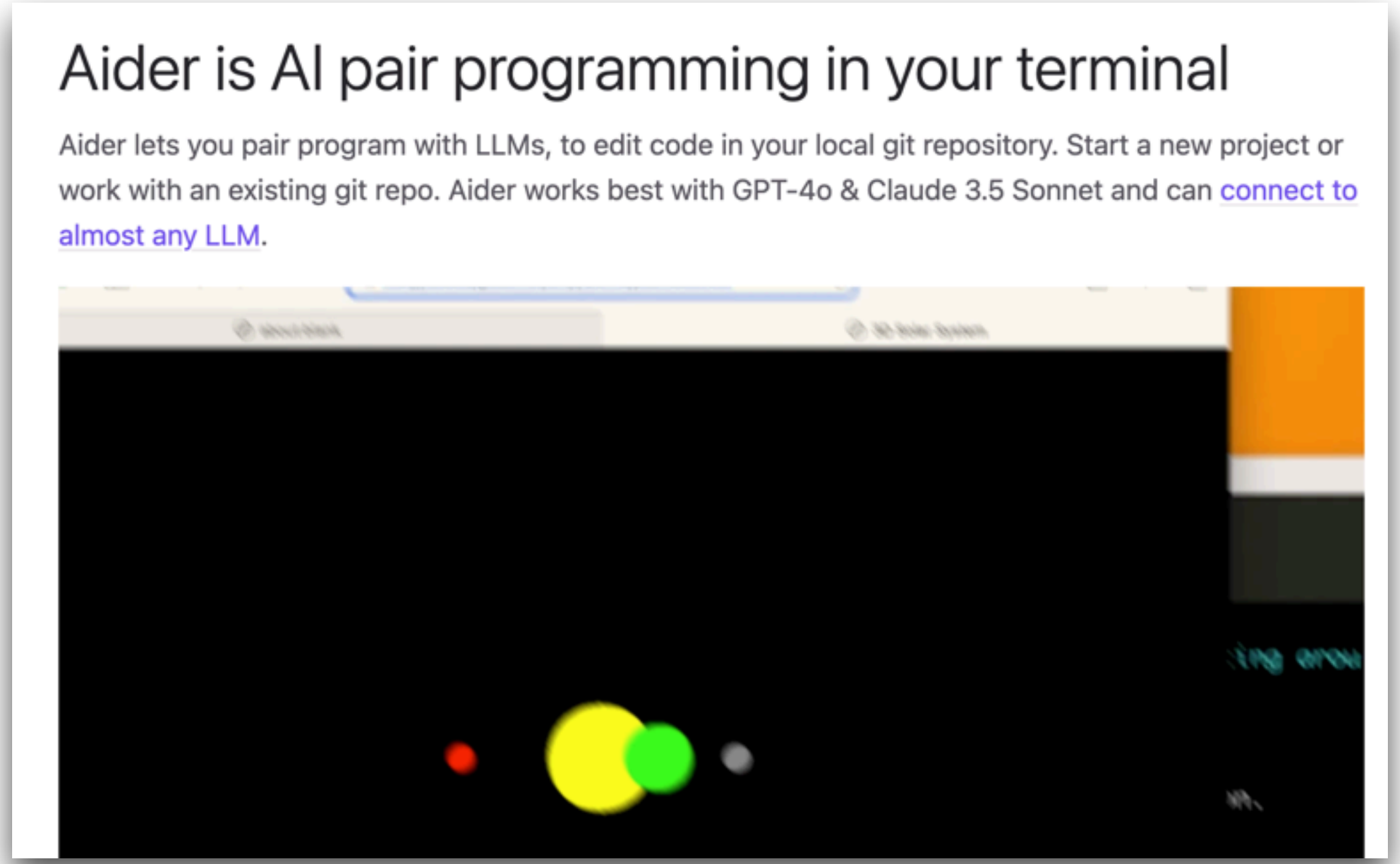
Workshop Record Expense

Requirement | Design | Develop | Testing | Deploy

Generative AI

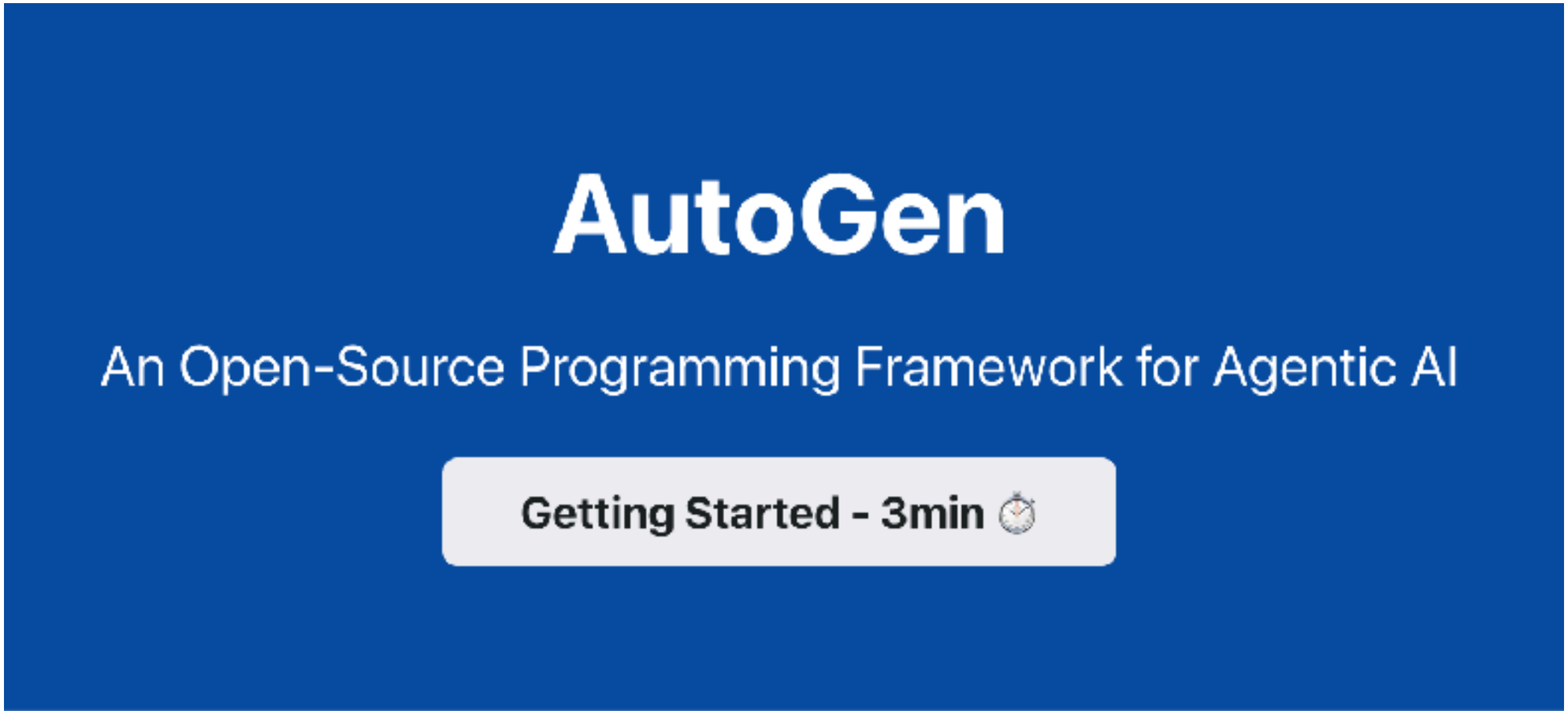
AI Agent Tools

Aider



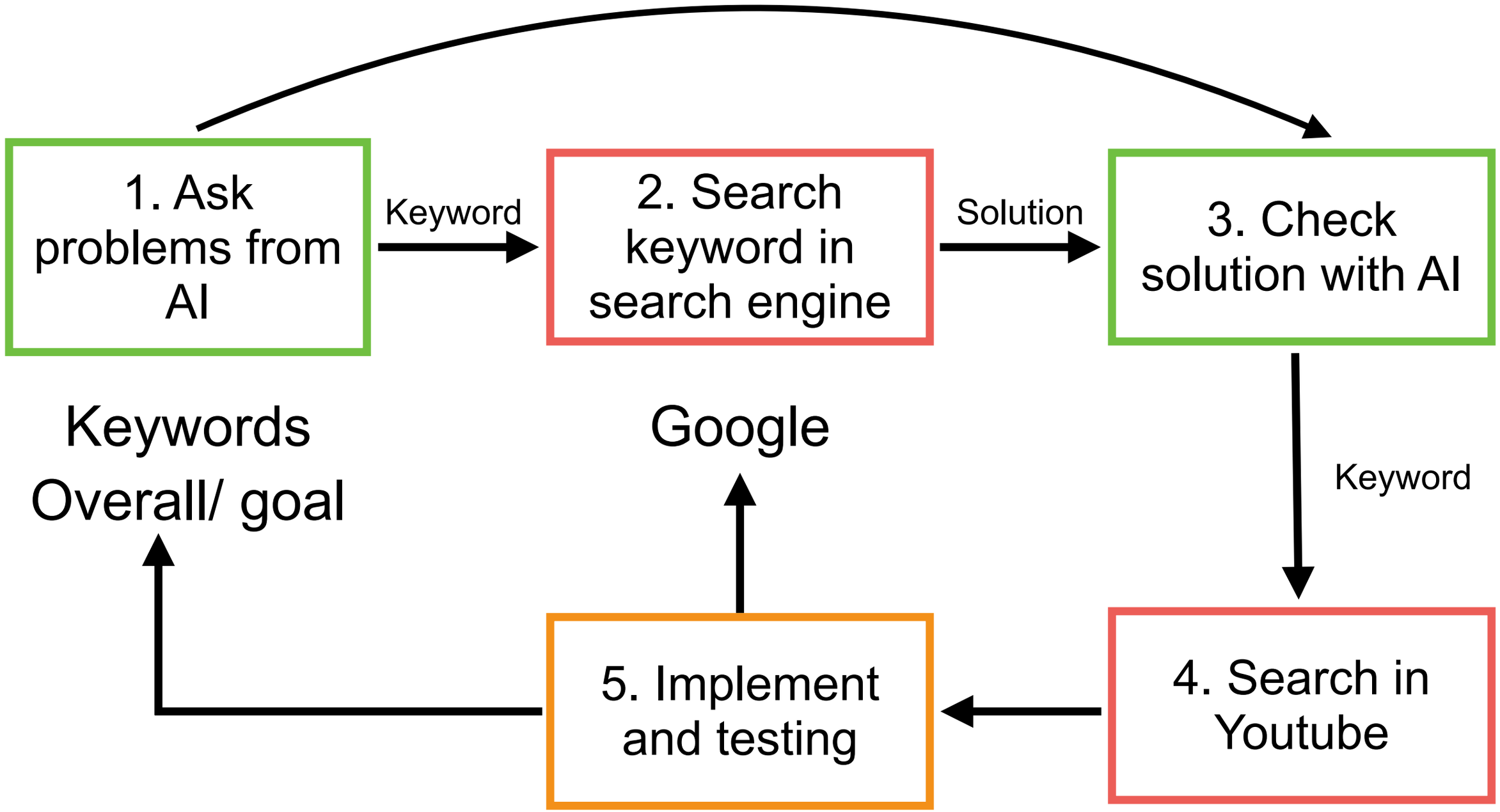
https://aider.chat/

Microsoft AutoGen



https://microsoft.github.io/autogen/

Learning Flow with AI



Q/A