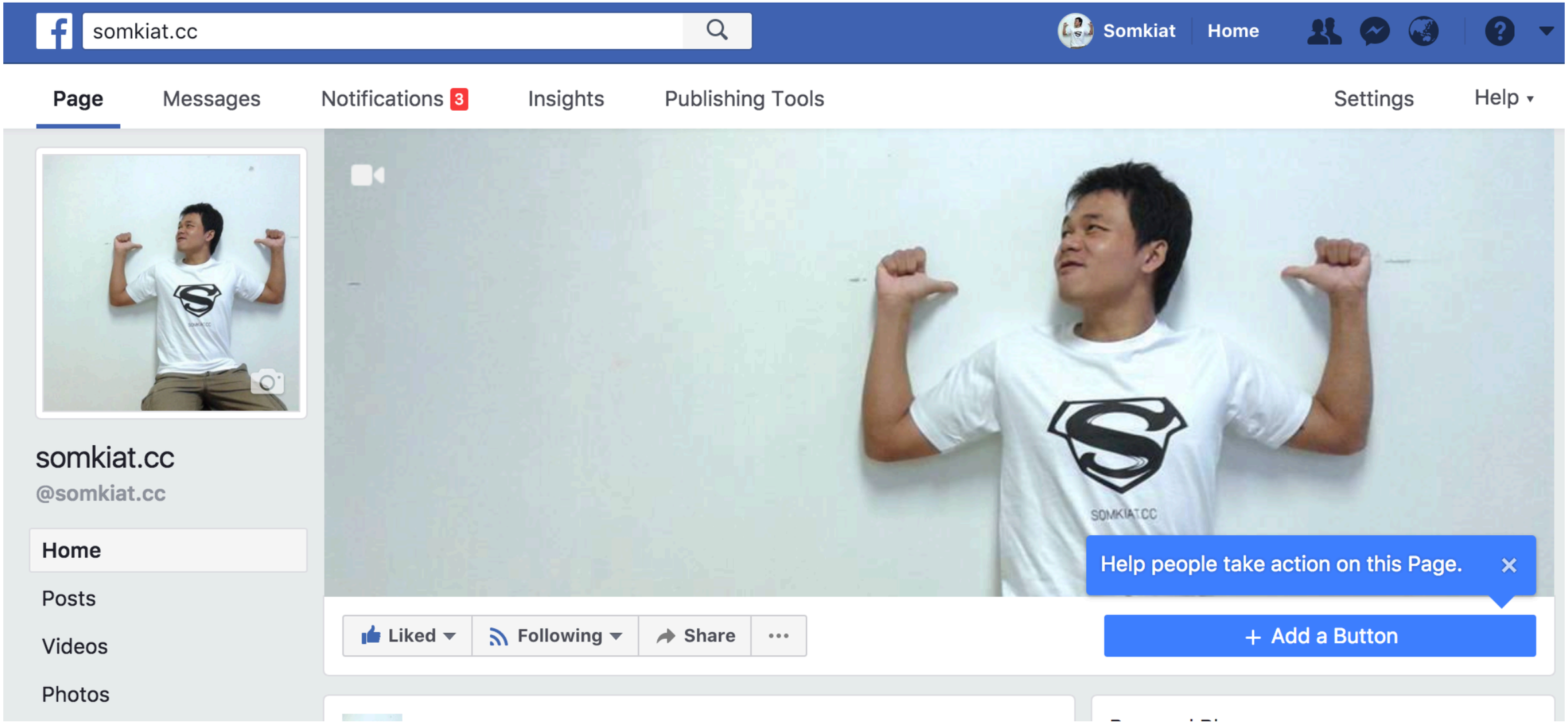
สยามชำนาญกิจ
SIAM CHAMNAN KIT

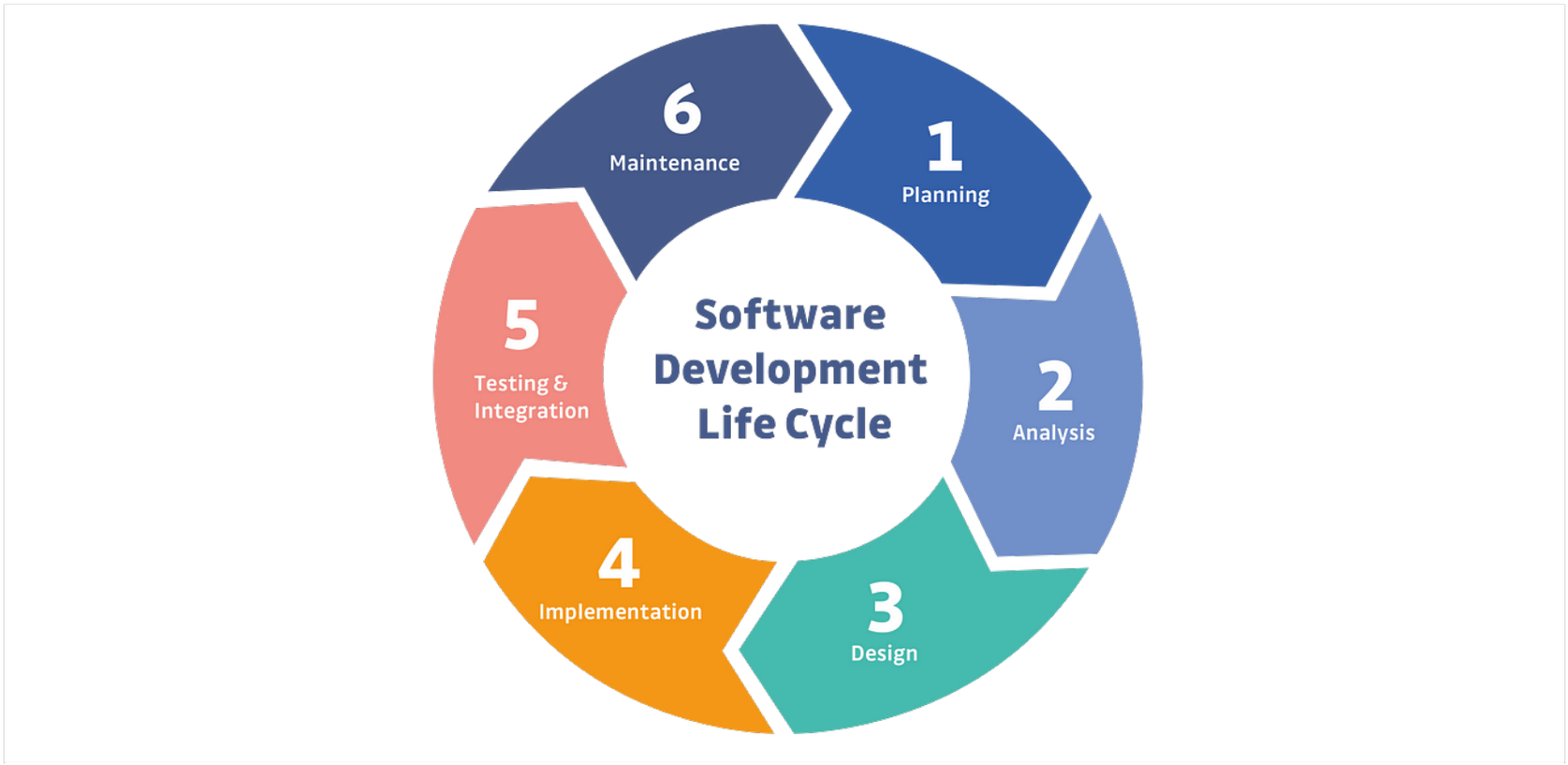
Gen AI for Software Development



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

Goals

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



Software Development

Requirement | Design | Develop | Testing | Deploy

Generative AI

Improve Productivity … (Replace human !!)

Learning Path

AI/ML

AI Model

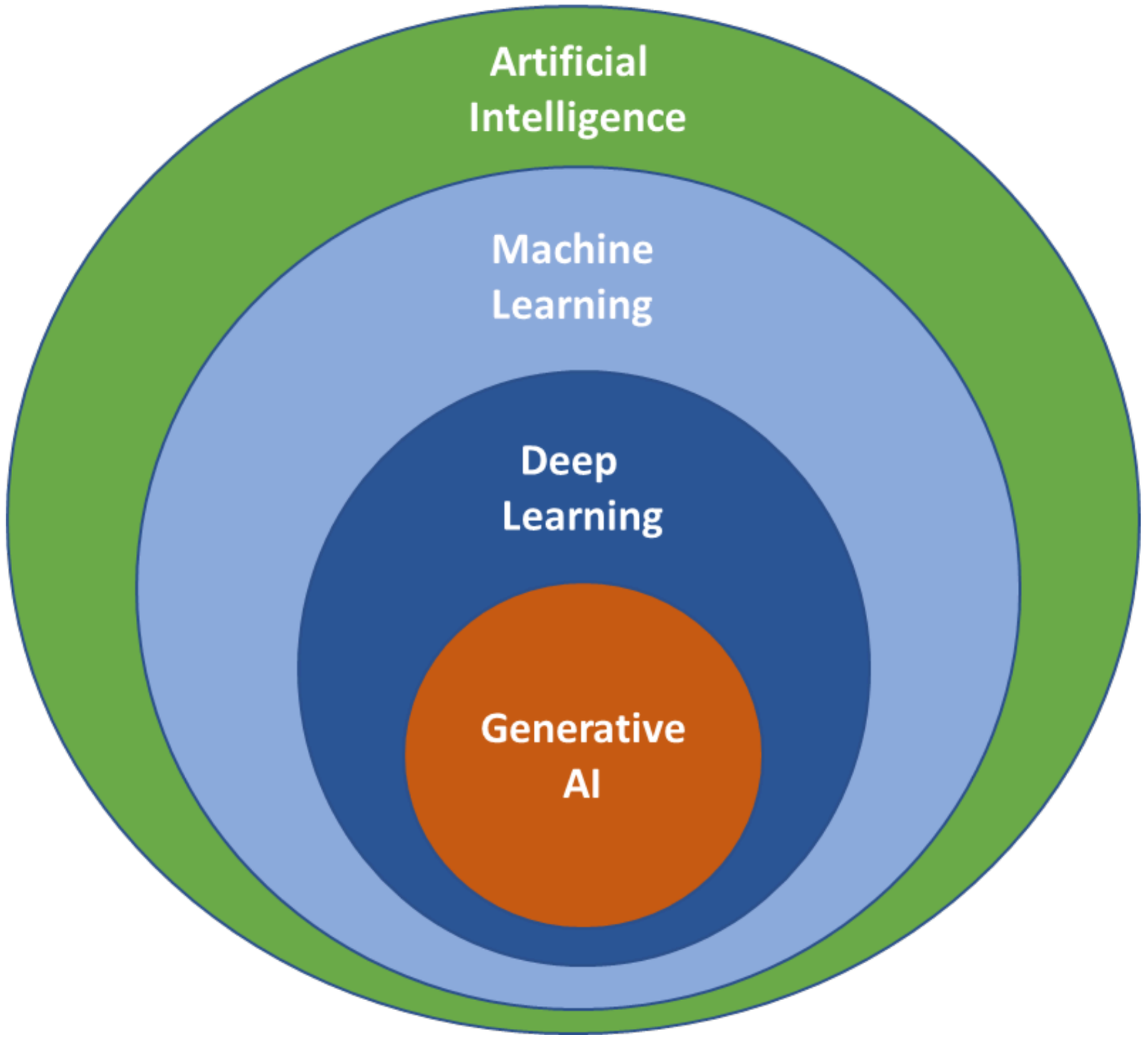
Prompt Engineer

AI in Software development

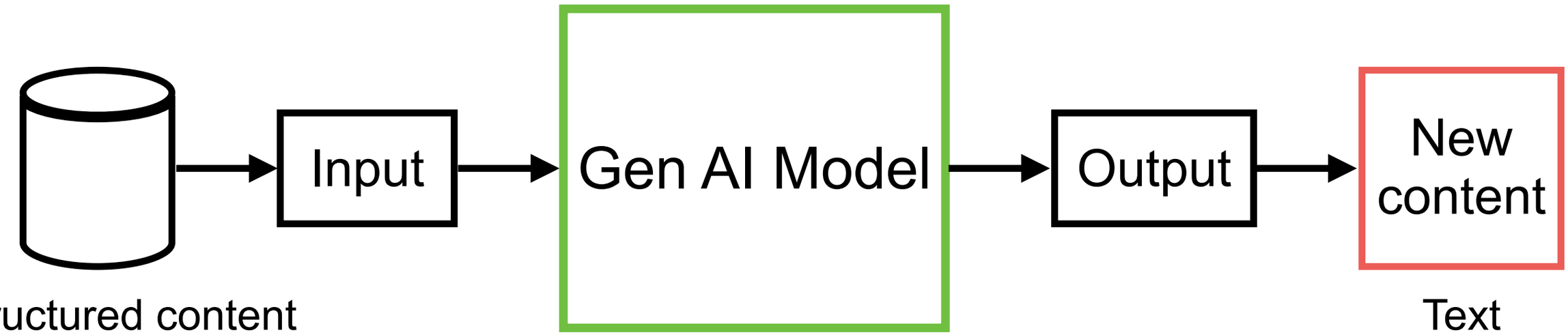
Develop AI/LLM app

RAG

Use cases and workshops



Generative AI



Unstructured content

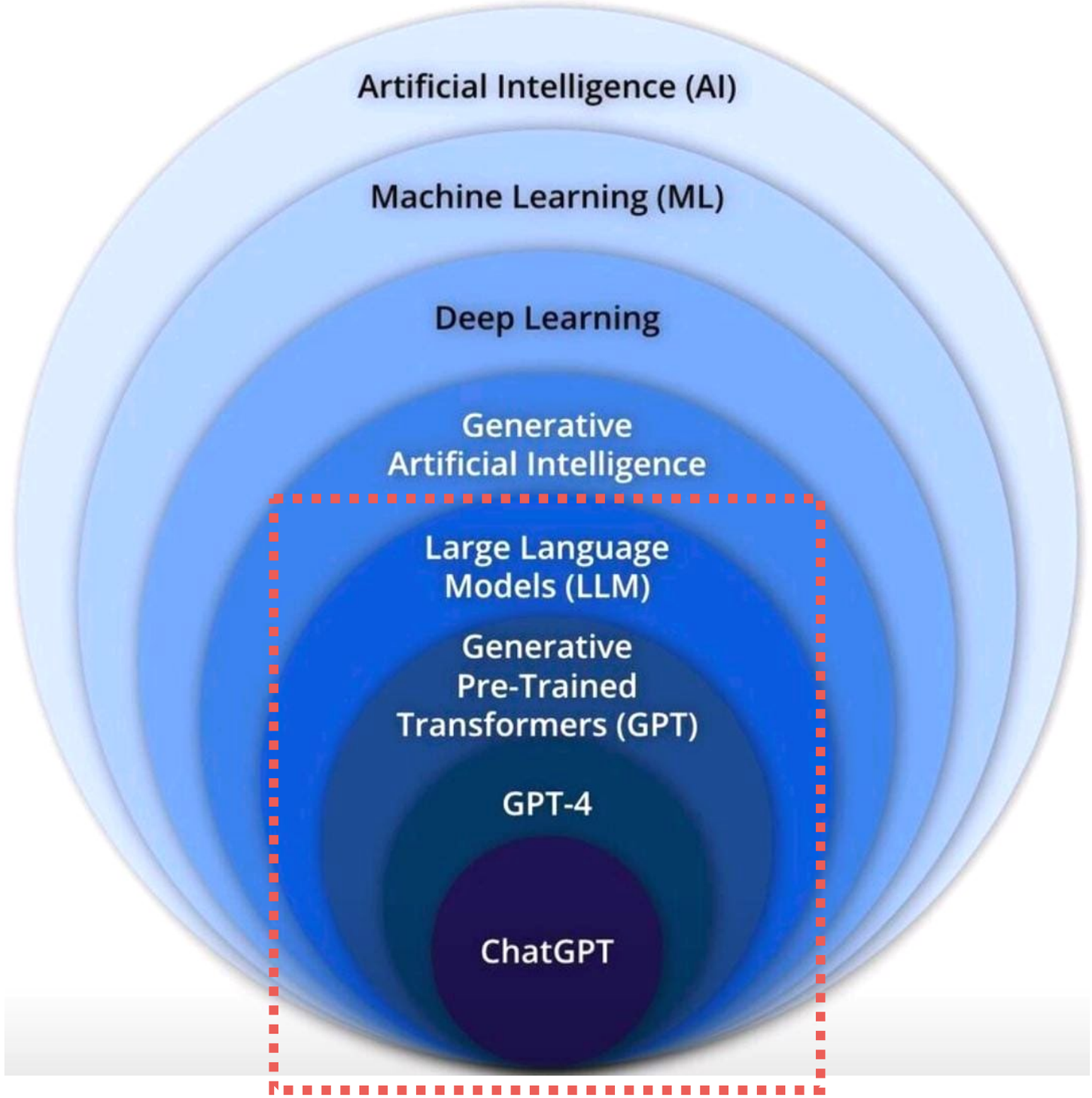
Learn patterns from Unstructured content

Text
Image
Video
3D
Task

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

Generative AI

LLMs	GANs	VAEs
Large Language Models	**Generative Adversarial Network**	**Variational Autoencoders**
Text generation	Image generation	Data compression
Code generation	Deep fake	Synthetic data generation
Chatbot	Art creation	Image reconstruction
Conversation AI	Simulate financial market	



Large Language Model (LLM)

Type of AI model
Process, understand
Generate human readable data

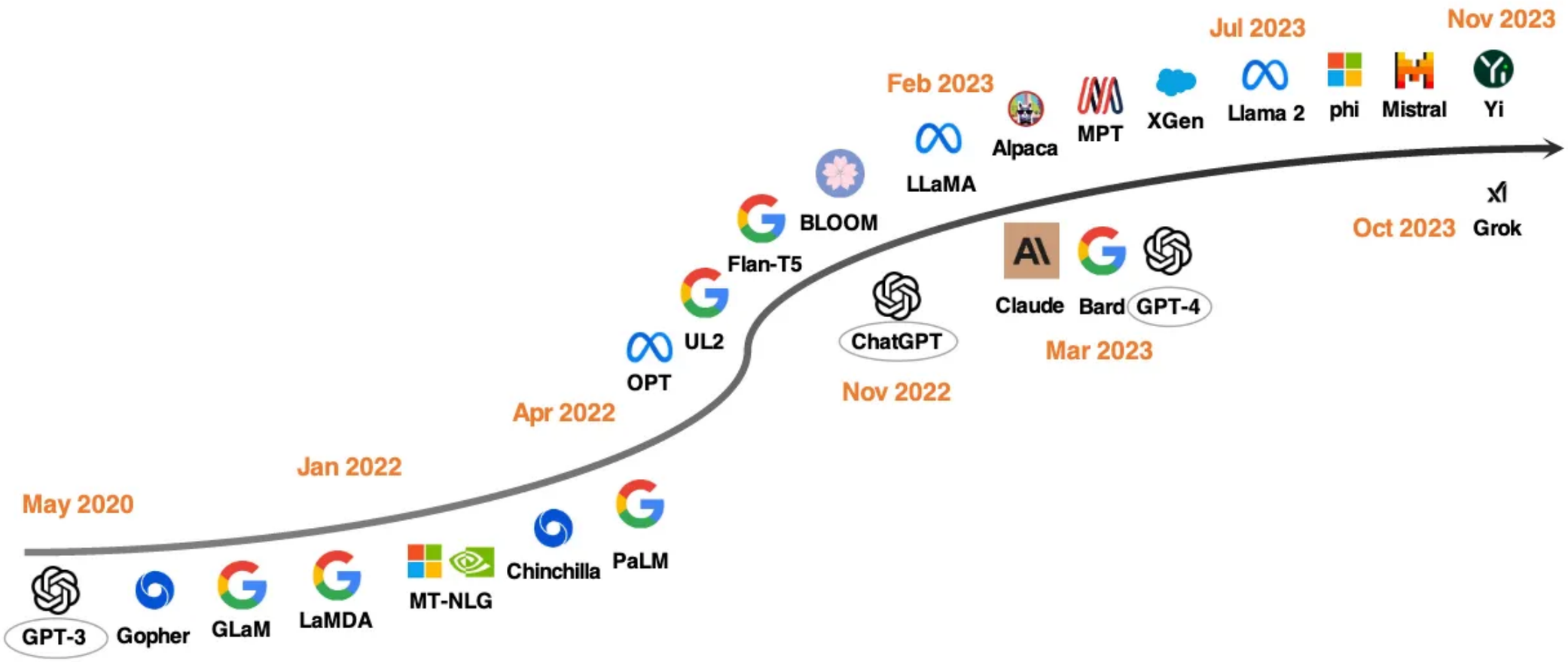
LLM

Training data !!

LLMs in industry

Model name	Company
Bidirectional Encoder Representation from Transformers (BERT)	Google AI
Generative Pre-trained transformer-5 (GPT-5)	OpenAI
Pathways Language Model-E (PaLM-E)	Google AI
BLOOM	NVIDIA AI
Llama 4	Facebook
Claude 4.5 Sonnet	Anthropic

LLM Development timeline



2025
GPT 4.1
O3, o4
Gemini 2.5
DeepSeek
Llama 4
Mistral
GLM

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

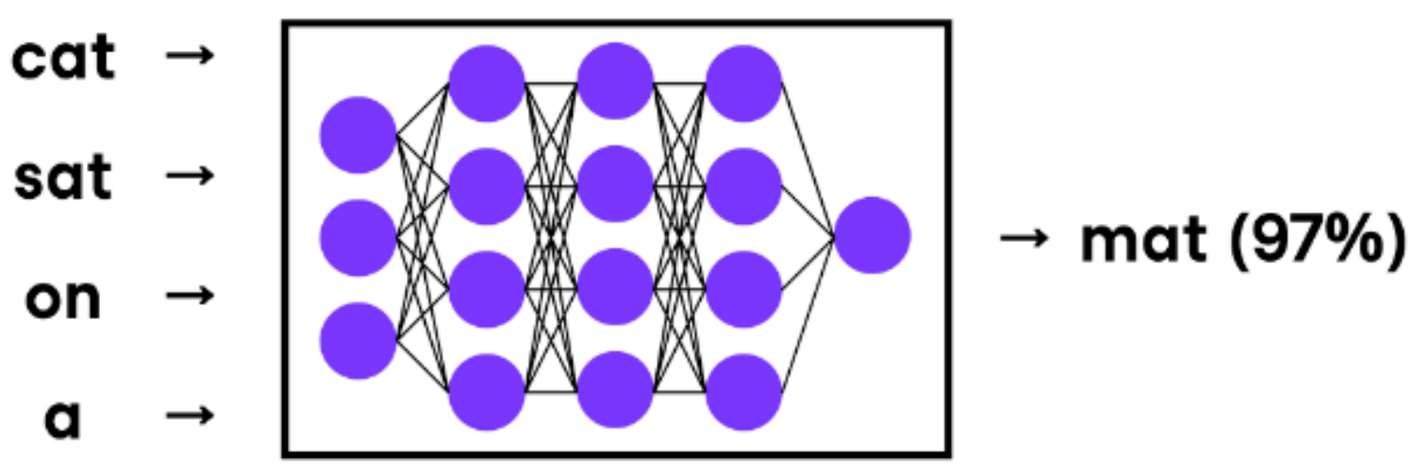
Let’s go !!

สีเหลือง

Large Language Model (LLM)

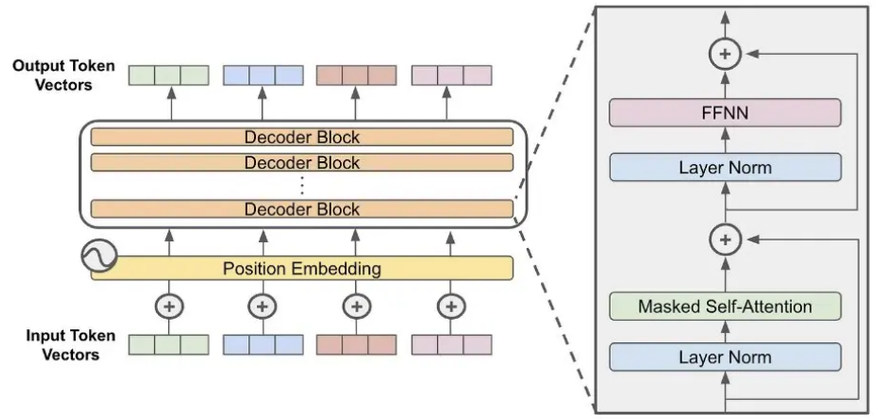
Neural network

Predicts the next word in a sequence

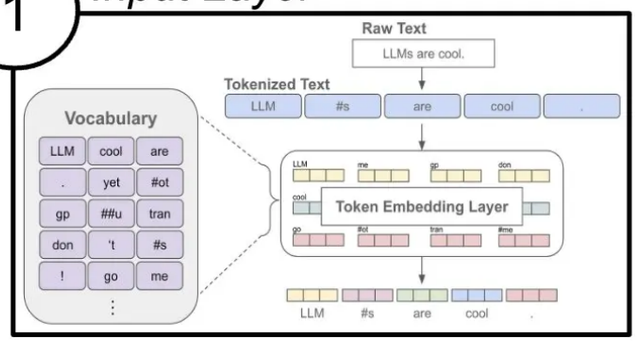


LLM components !!

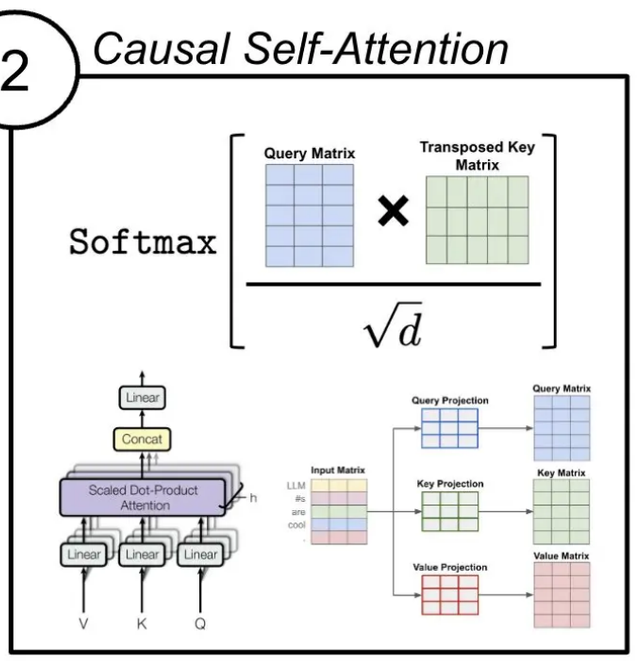
Components of the Decoder-only Transformer



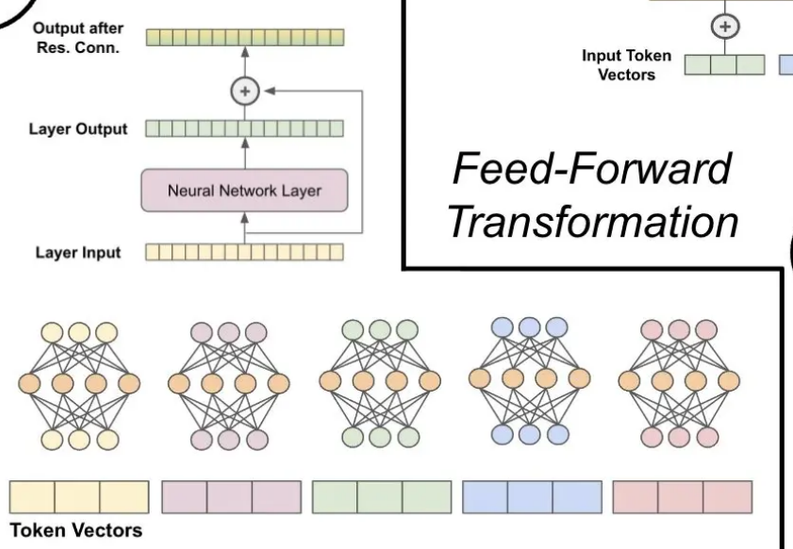
1 *Input Layer*



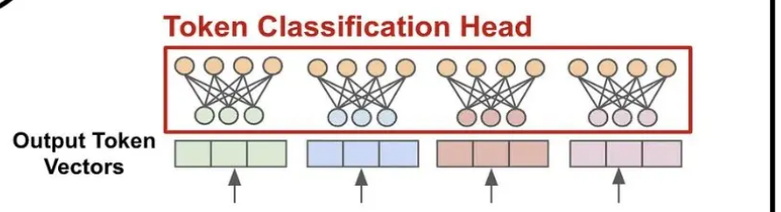
2 *Causal Self-Attention*



3 *Feed-Forward Transformation*



4 *Classification Head*

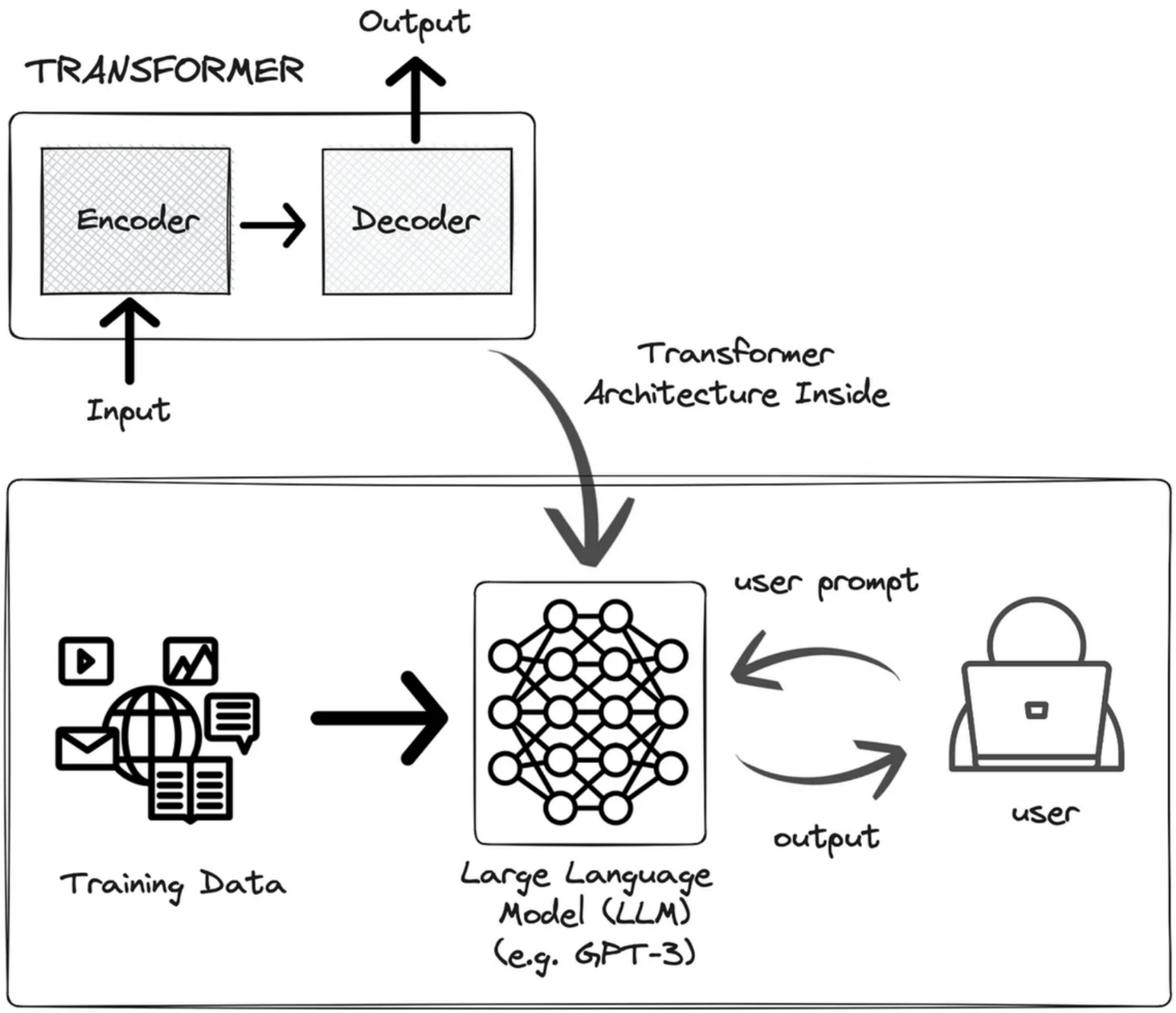


4 *Transformer Block*

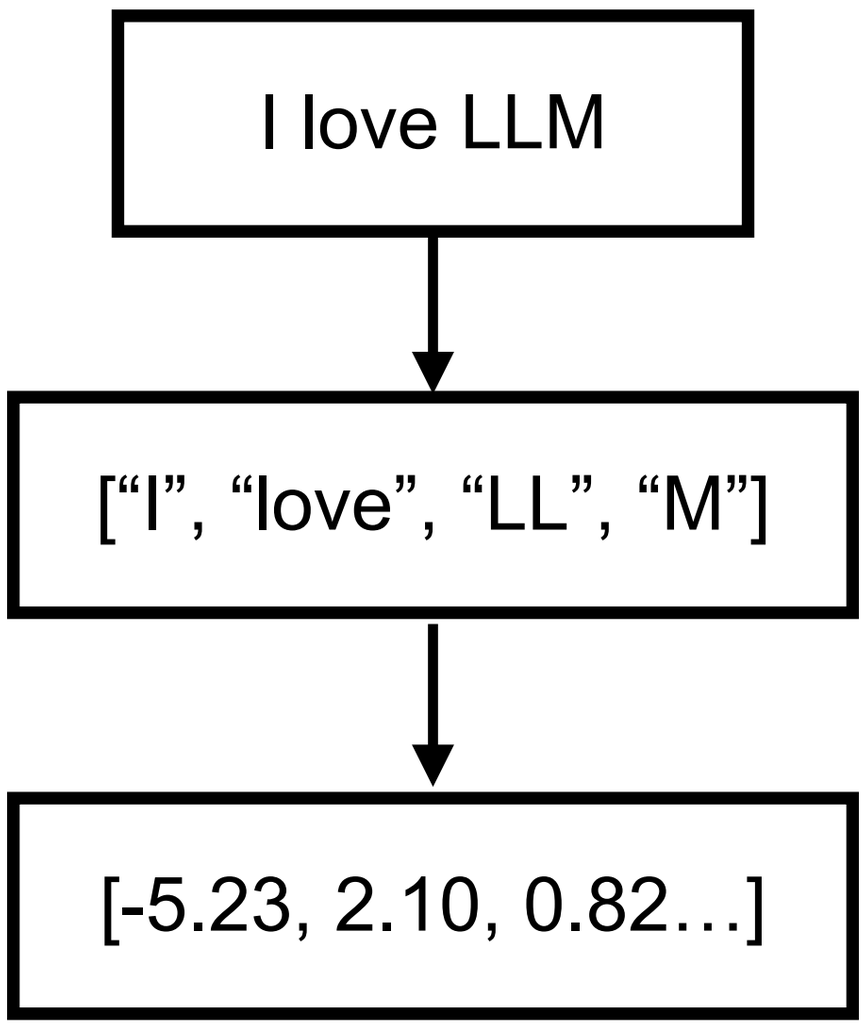
```python
from torch import nn

class Block(nn.Module):
    def __init__(
        self,
        d,
        H,
        T,
        bias=False,
        dropout=0.2,
    ):
        """
        d: size of embedding dimension
        H: number of attention heads
        T: maximum length of input sequences (in tokens)
        bias: whether or not to use bias in linear layers
        dropout: probability of dropout
        """
        super().__init__()
        self.ln_1 = nn.LayerNorm(d)
        self.attn = CausalSelfAttention(d, H, T, bias, dropout)
        self.ln_2 = nn.LayerNorm(d)
        self.ffnn = FFNN(d, bias, dropout)

    def forward(self, x):
        x = x + self.attn(self.ln_1(x))
        x = x + self.ffnn(self.ln_2(x))
        return x
```

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

Transformer inside



Transformer process

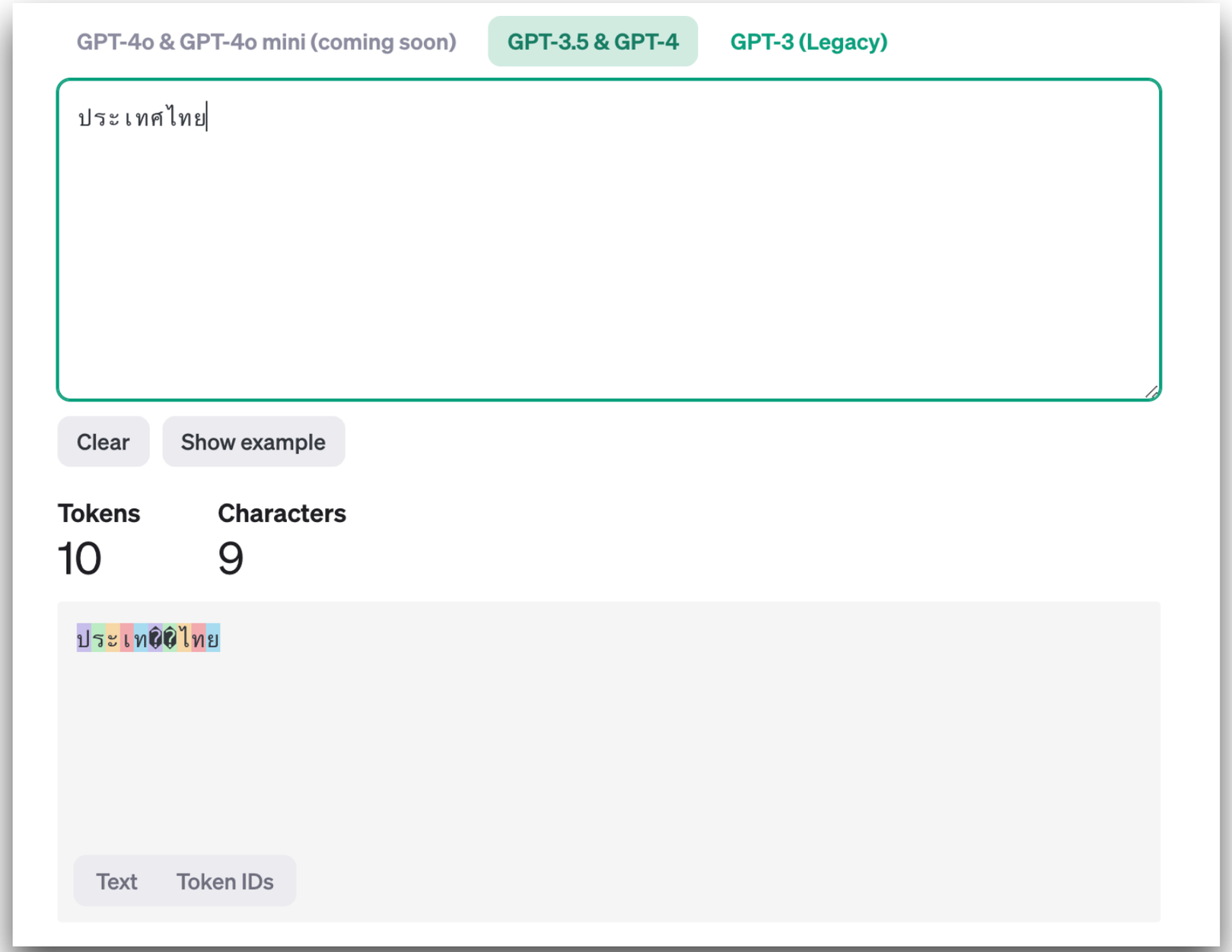


Sentence

Tokens

Embedding

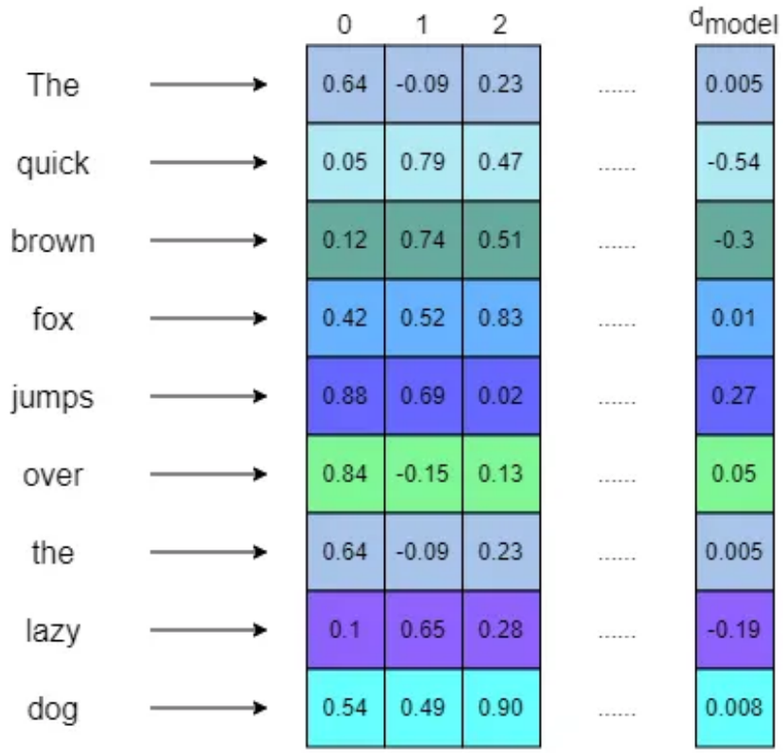
OpenAI Tokenizer

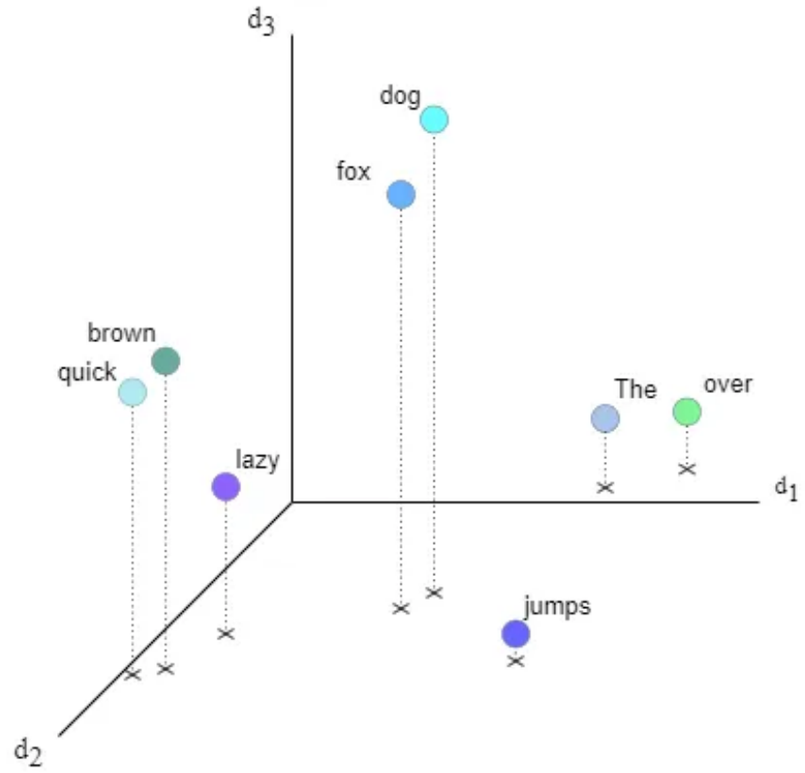


https://platform.openai.com/tokenizer

Embedding ?

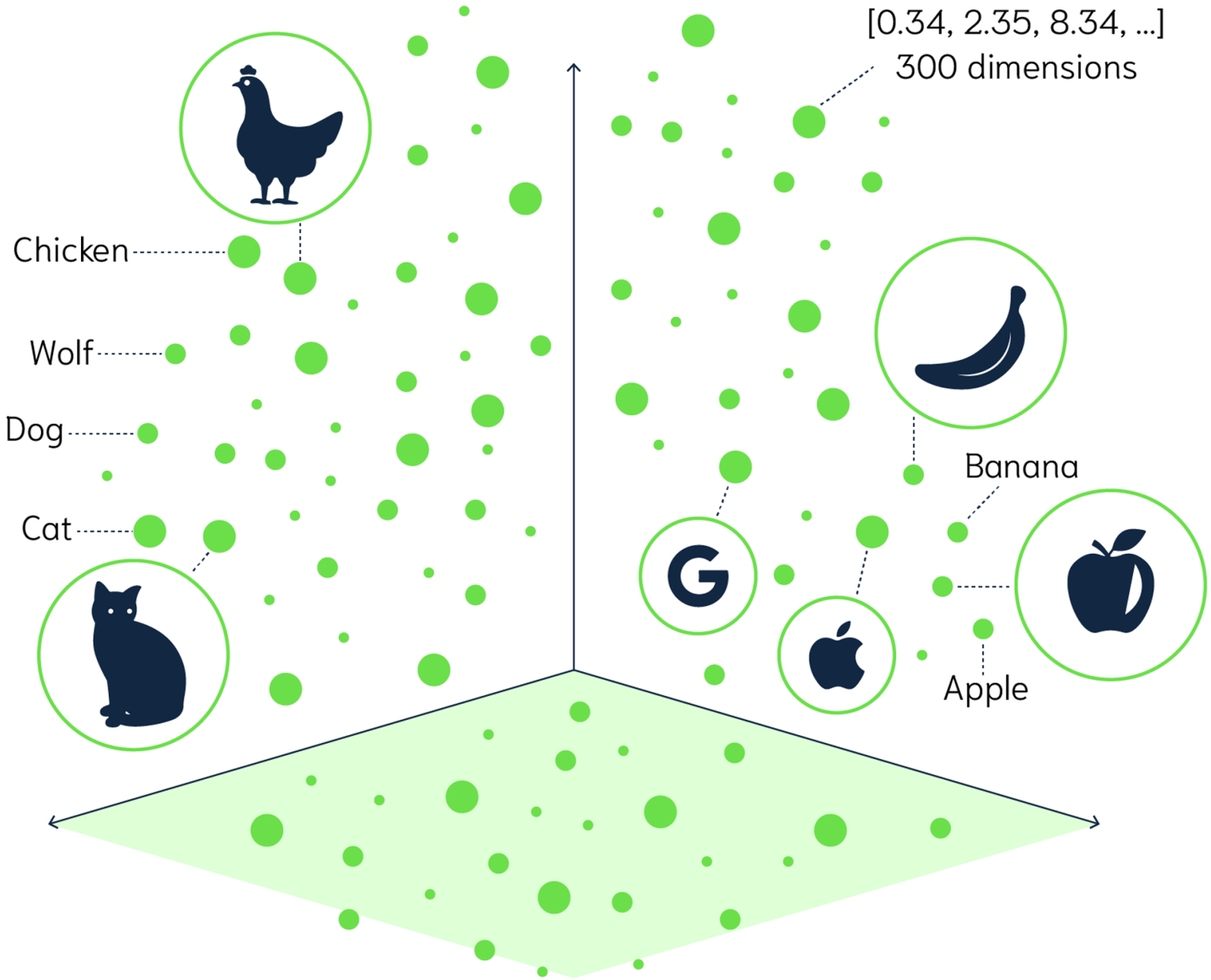
Map items of **unstructured data** to high-dimensional real vectors





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

Visual of Vector space

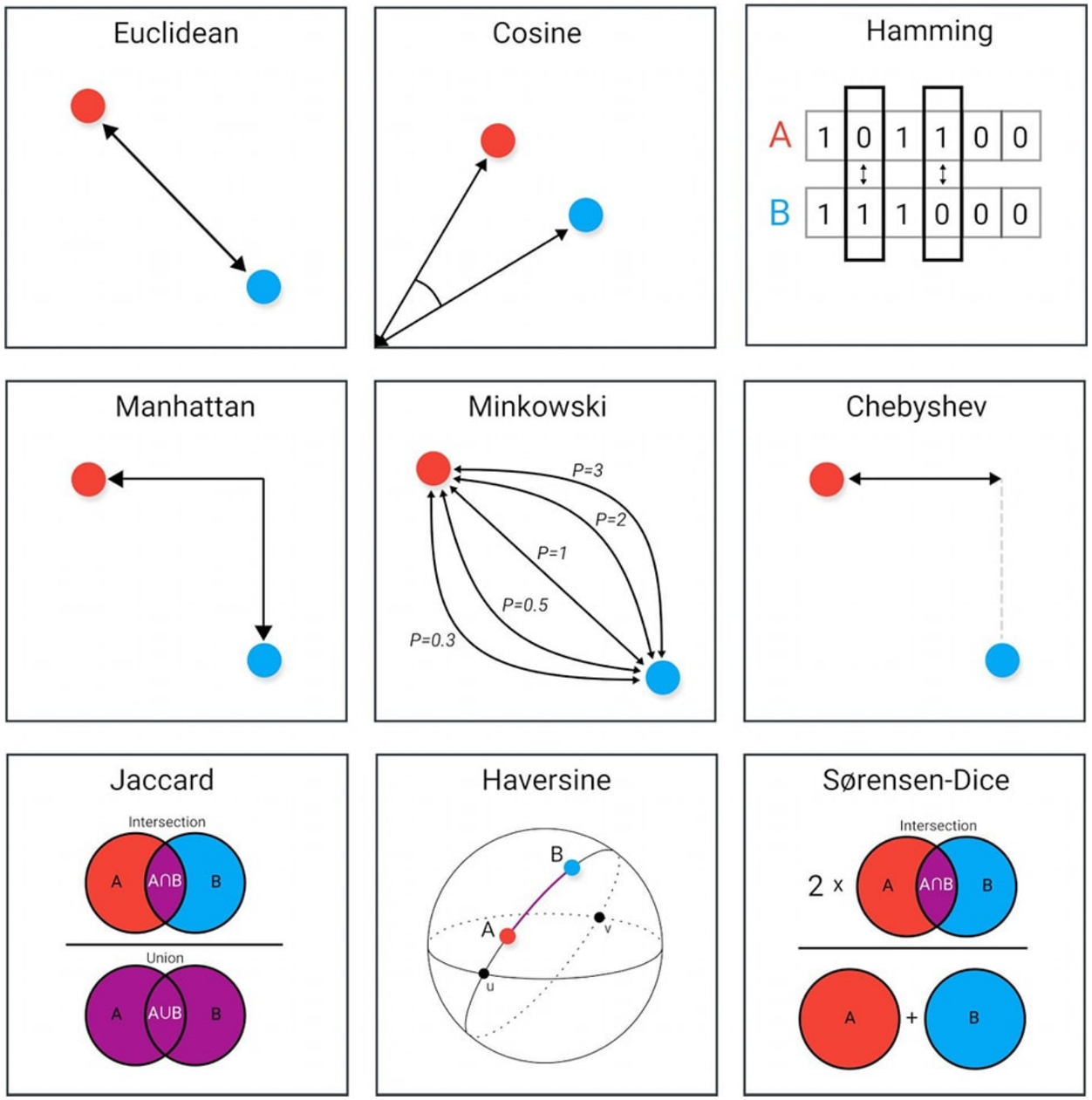


Embedding Leaderboard

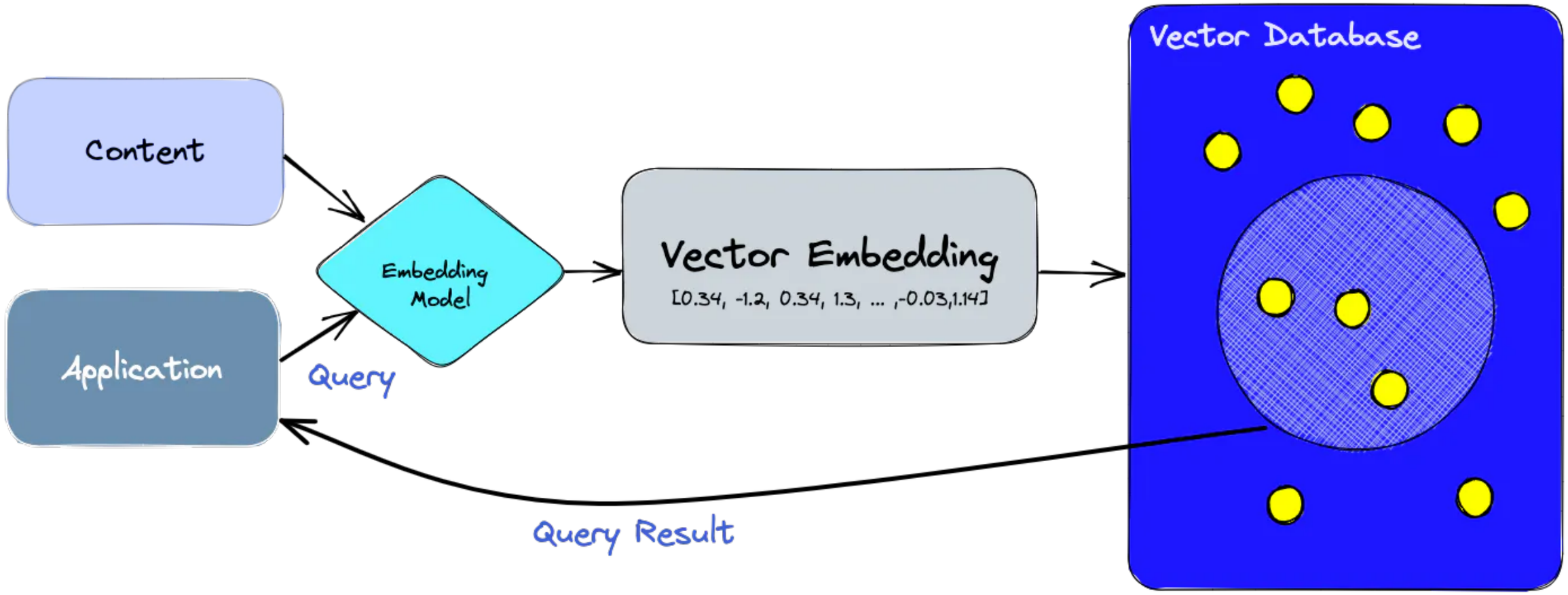
Rank (Bor…	Model	Zero-shot	Memory U…	Number of P…	Embedding D…	Max Tokens	Mean (T…	Mean (TaskT…	Bitext …	Classific
1	llama-embed-nemotron-8b	99%	28629	7B	4096	32768	69.46	61.09	**81.72**	73.21
2	Qwen3-Embedding-8B	99%	28866	7B	4096	32768	**70.58**	**61.69**	80.89	**74.00**
3	gemini-embedding-001	99%	Unknown	Unknown	3072	2048	68.37	59.59	79.28	71.82
4	Qwen3-Embedding-4B	99%	15341	4B	2560	32768	69.45	60.86	79.36	72.33
5	Qwen3-Embedding-0.6B	99%	2272	595M	1024	32768	64.34	56.01	72.23	66.83
6	gte-Qwen2-7B-instruct	⚠ NA	29040	7B	3584	32768	62.51	55.93	73.92	61.55
7	Linq-Embed-Mistral	99%	13563	7B	4096	32768	61.47	54.14	70.34	62.24
8	multilingual-e5-large-instruct	99%	1068	560M	1024	514	63.22	55.08	80.13	64.94
9	embeddinggemma-300m	99%	578	307M	768	2048	61.15	54.31	64.40	60.90
10	SFR-Embedding-Mistral	96%	13563	7B	4096	32768	60.90	53.92	70.00	60.02
11	text-multilingual-embedding-002	99%	Unknown	Unknown	768	2048	62.16	54.25	70.73	64.64

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

Distance measure in Data Science

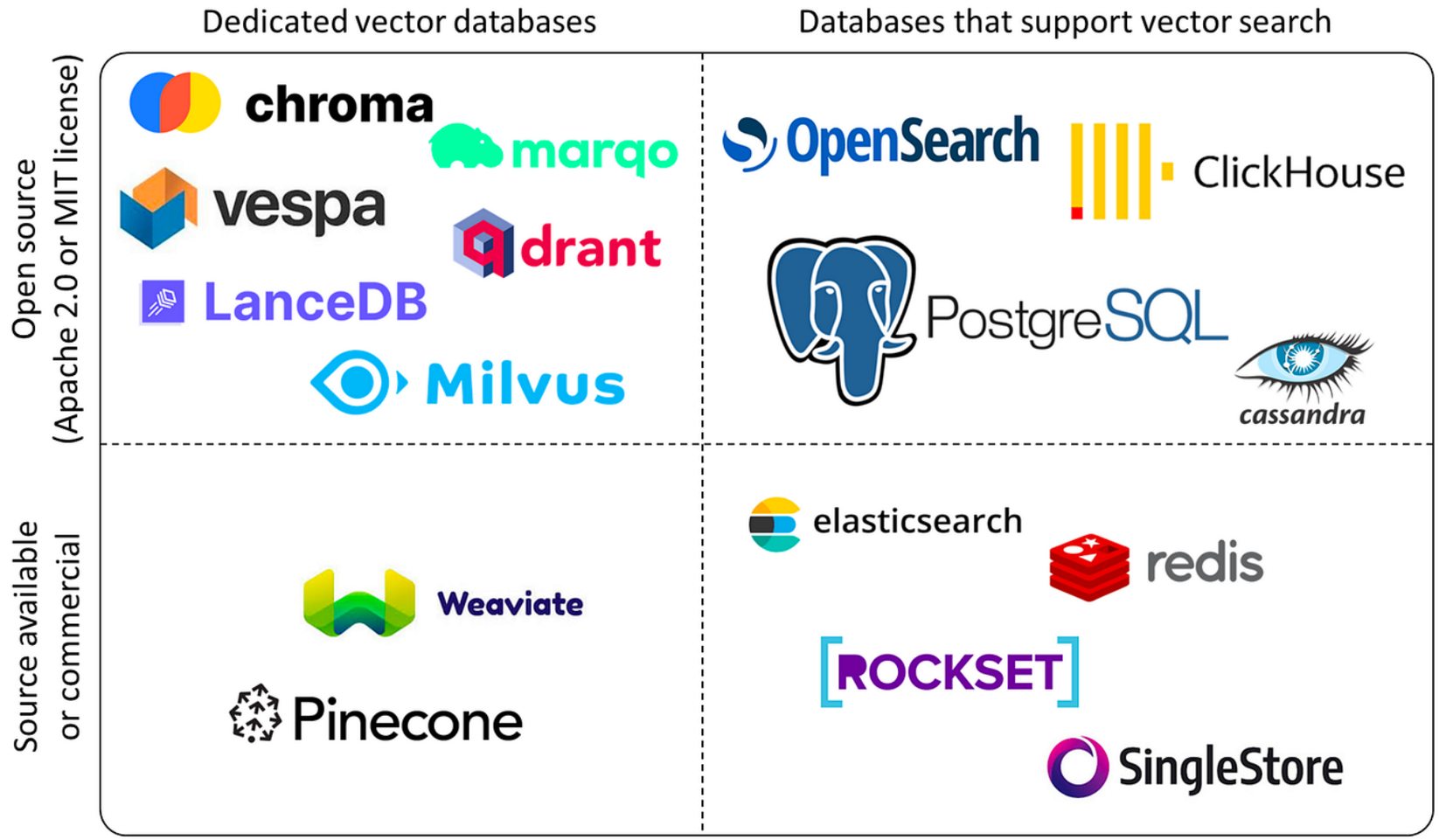


Store data in Vector Database

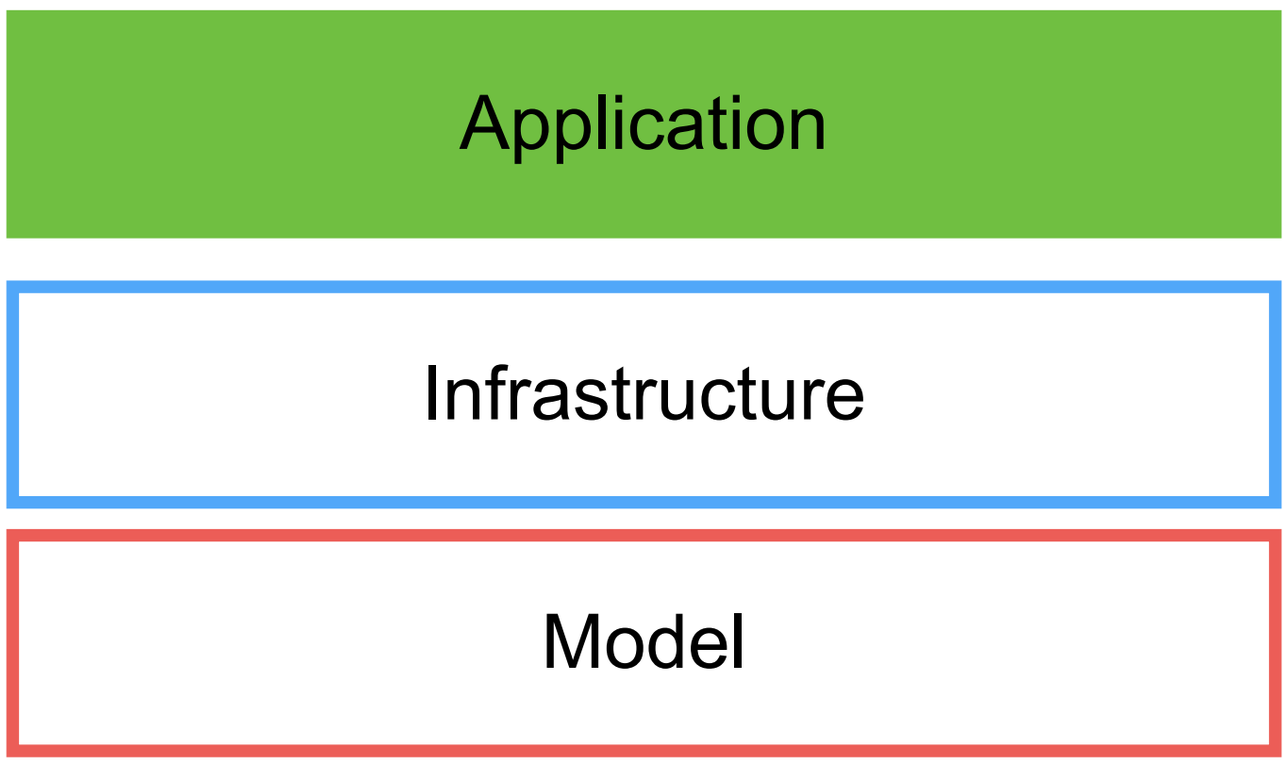


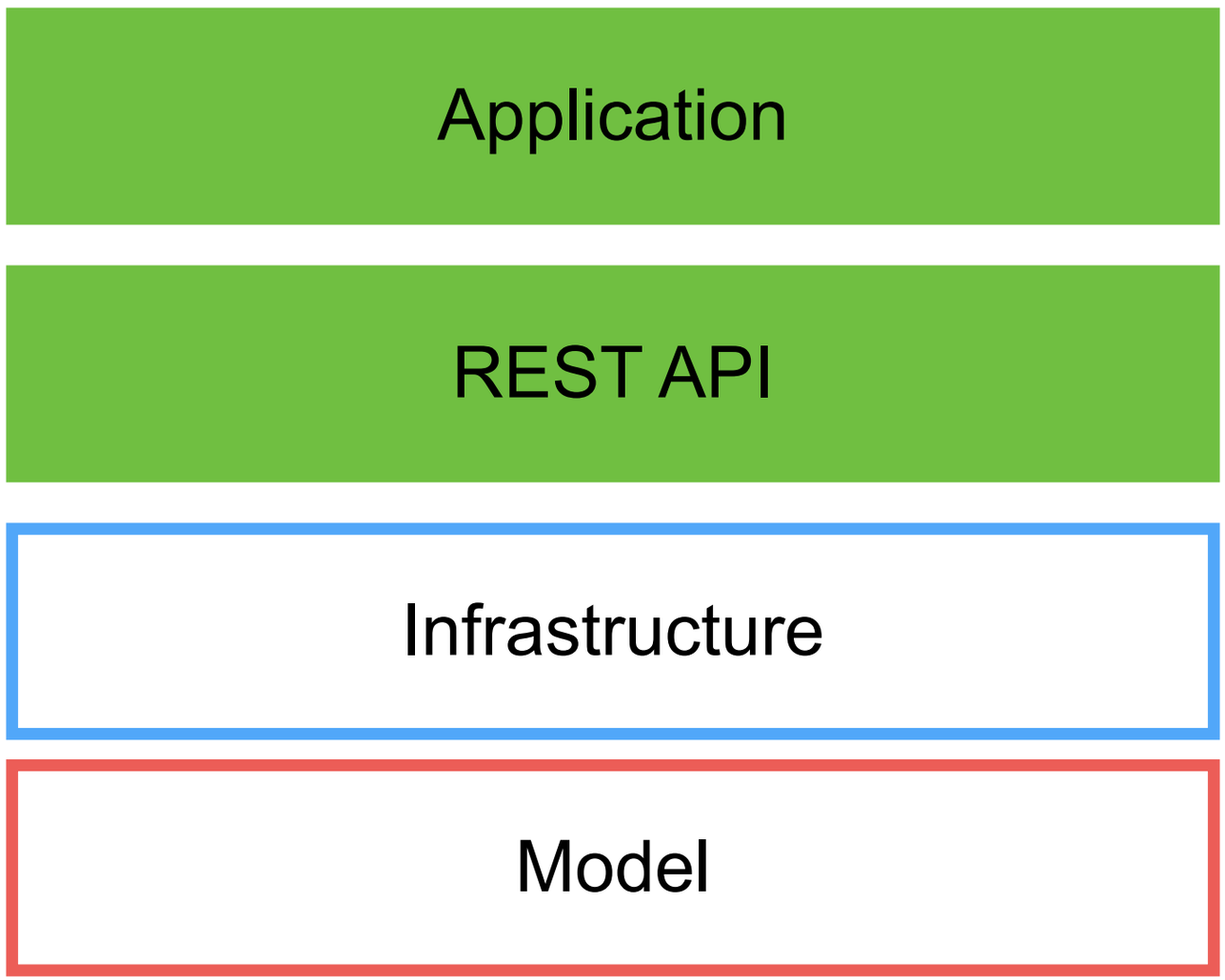
Vector Database ?

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

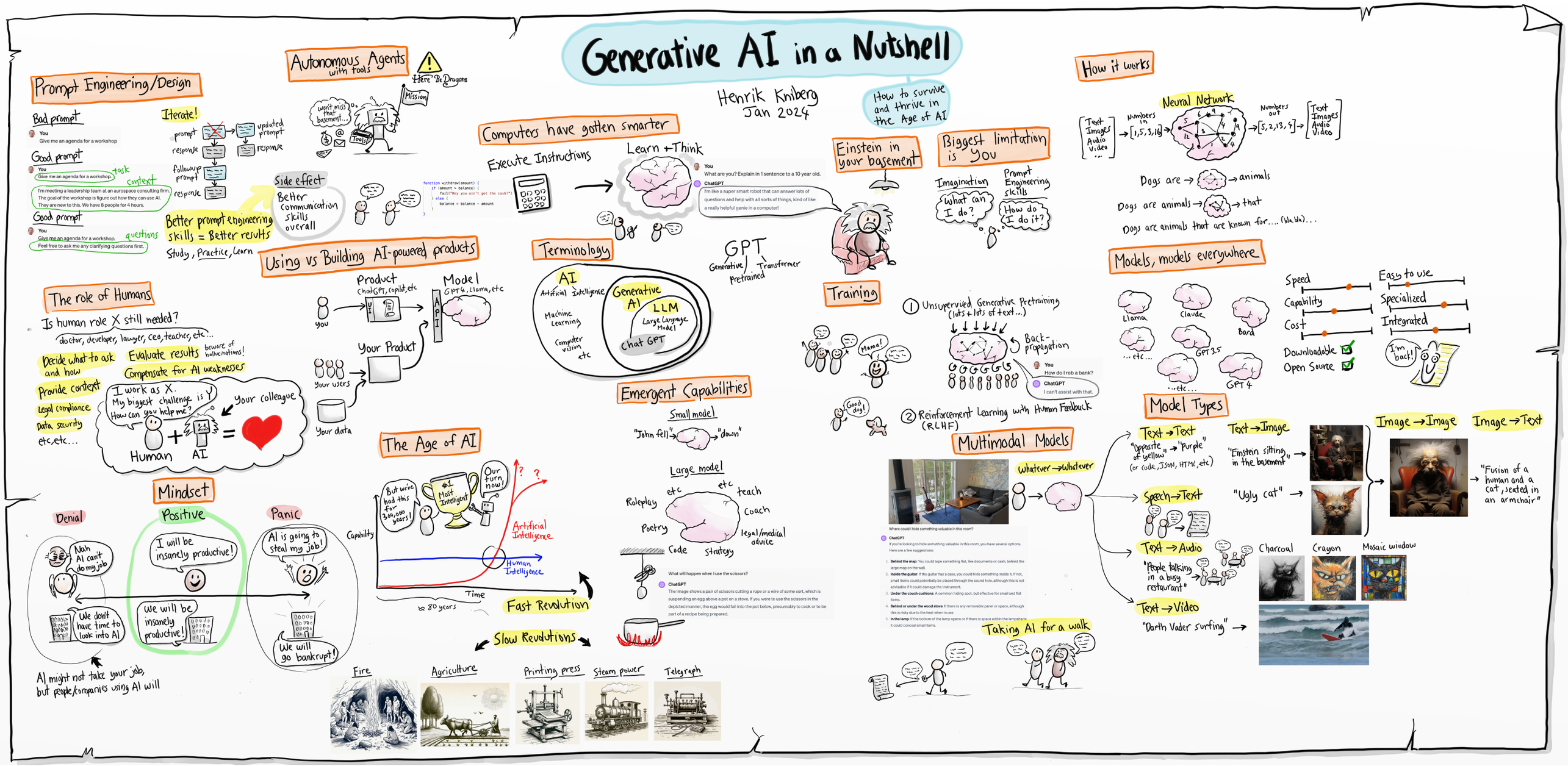


https://towardsdatascience.com/transformers-in-depth-part-1-introduction-to-transformer-models-in-5-minutes-ad25da6d3cca



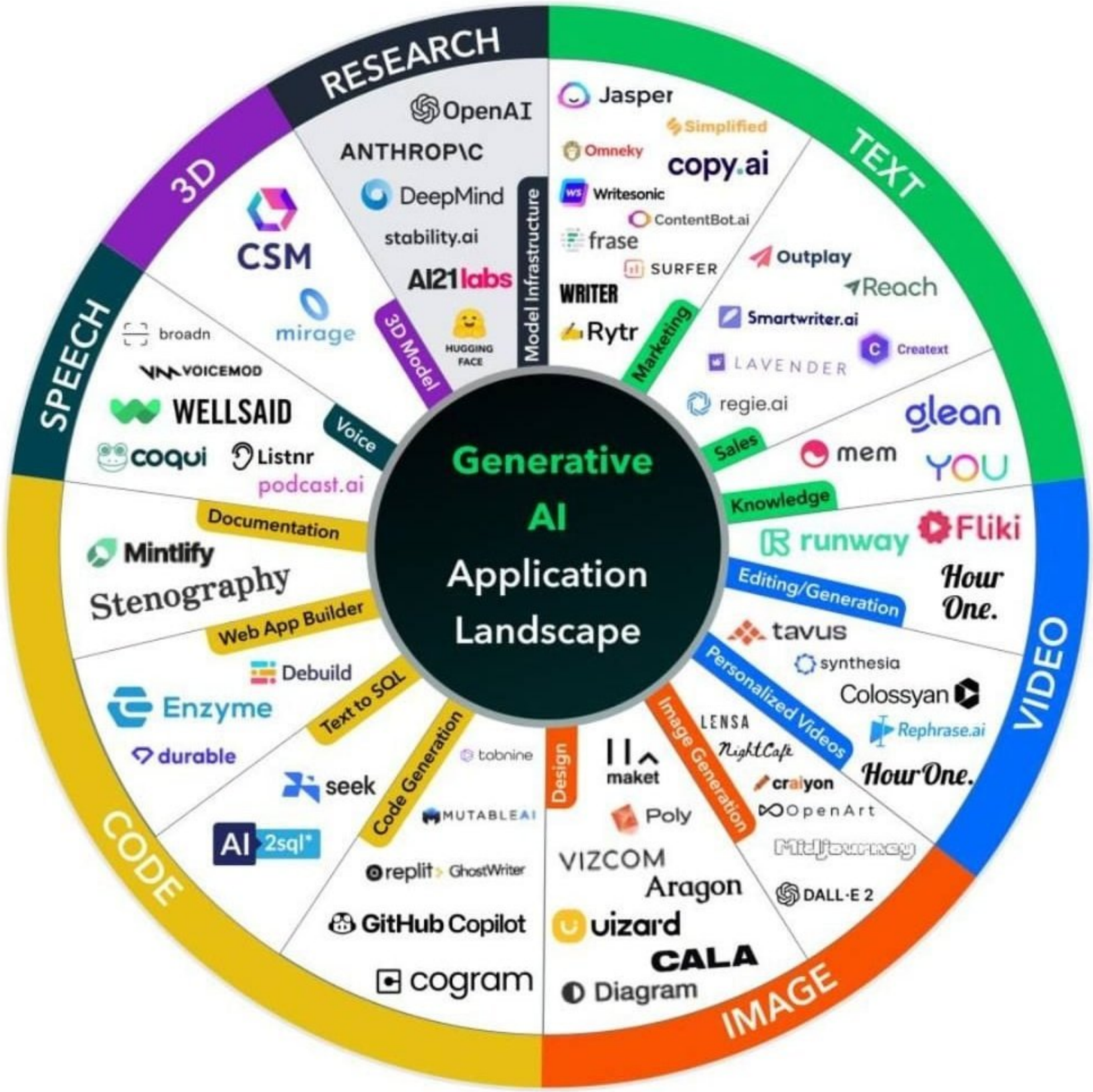


Generative AI in Nutshell



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

Application
Tool chains



Tool chains category

Assist tasks

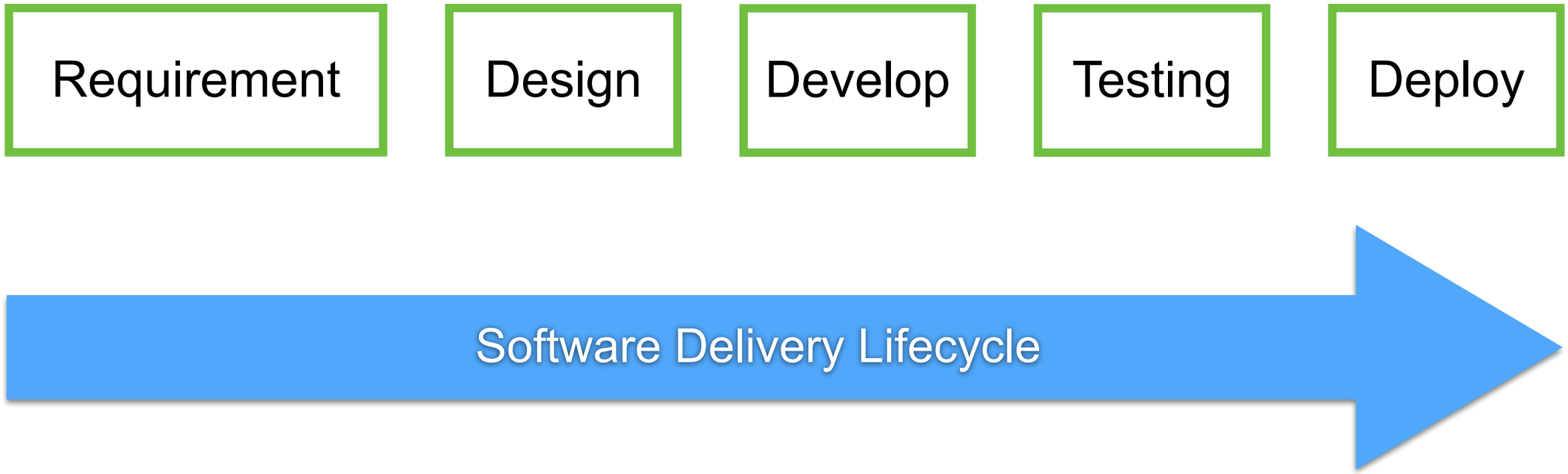
Interaction modes

Prompt composition

Properties of model

Assist tasks

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



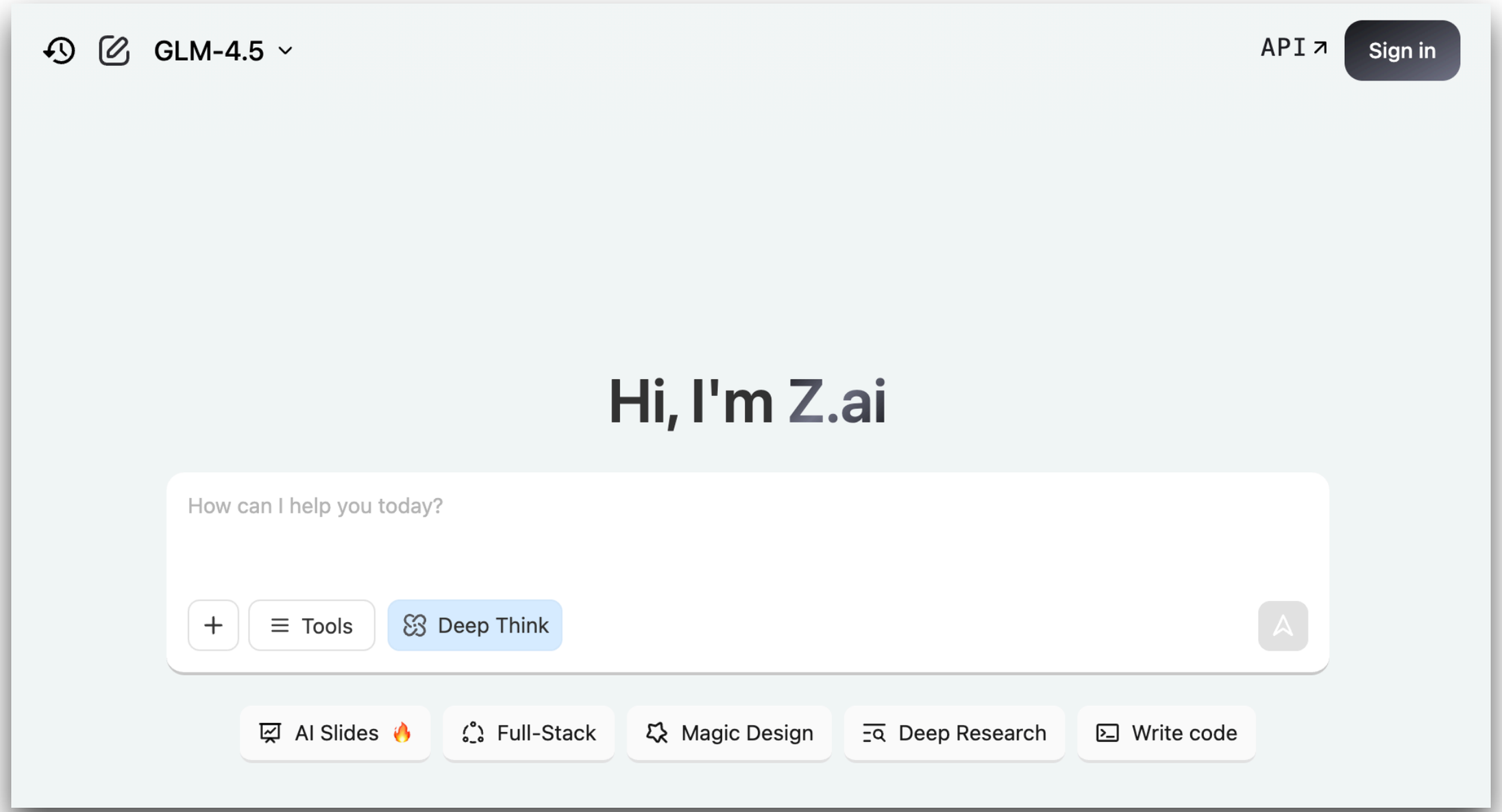
Interaction modes

Chat interfaces

In-line assistance (typing in code editor)

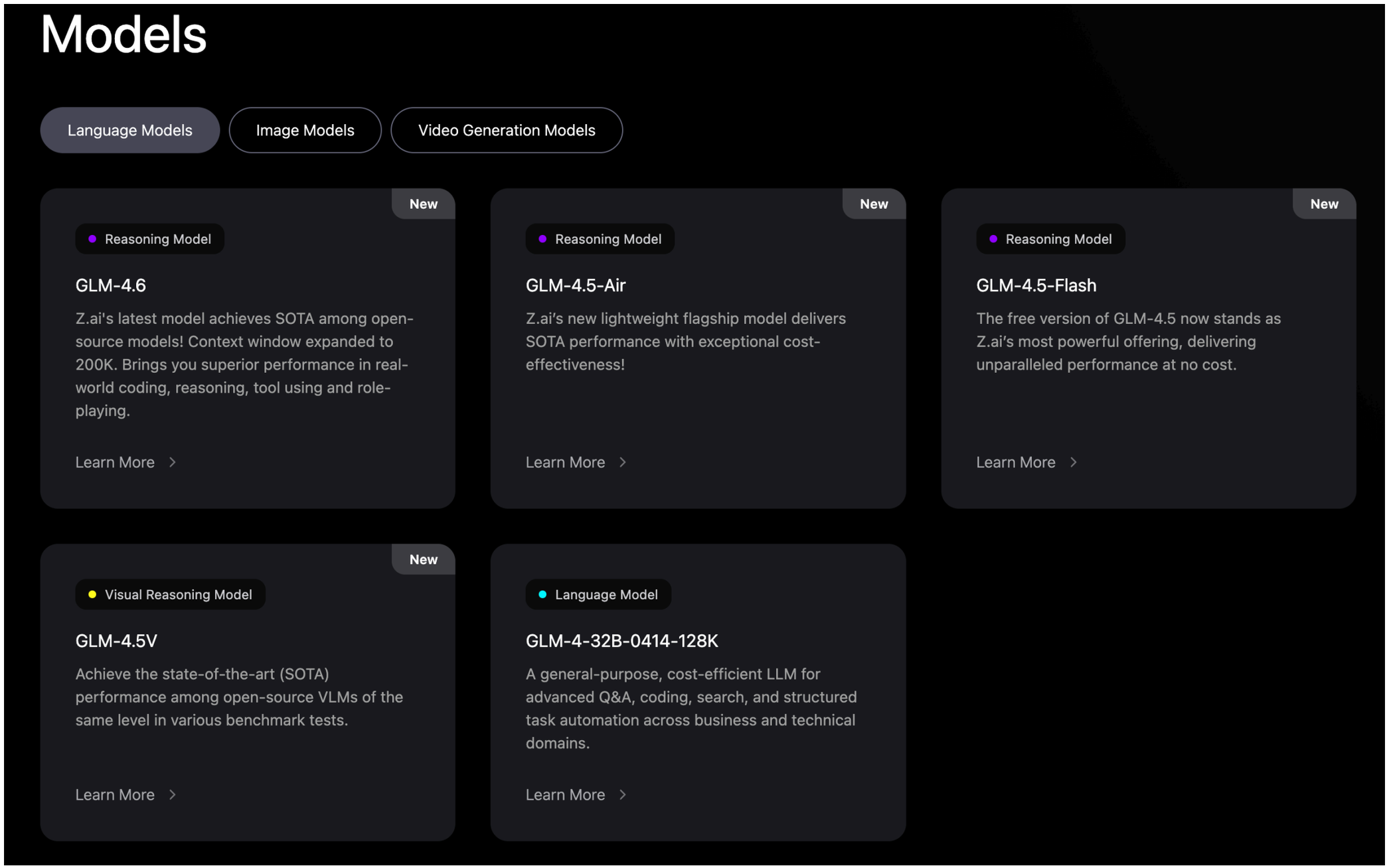
CLI (command-line interface)

Z.ai



https://z.ai/

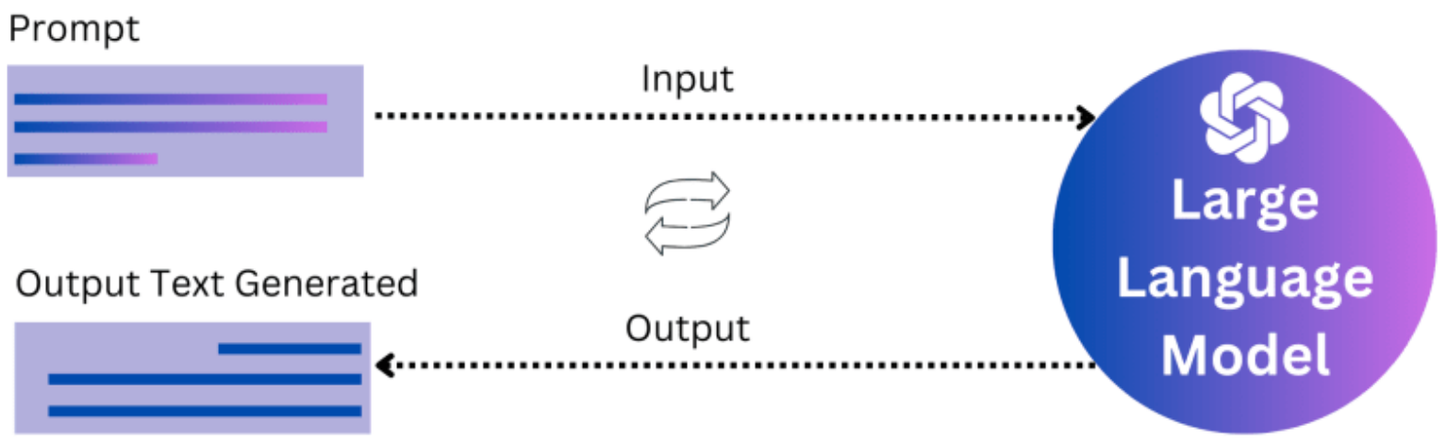
Models



https://z.ai/

Prompt composition

Prompt engineering
Compose prompts from user inputs and context



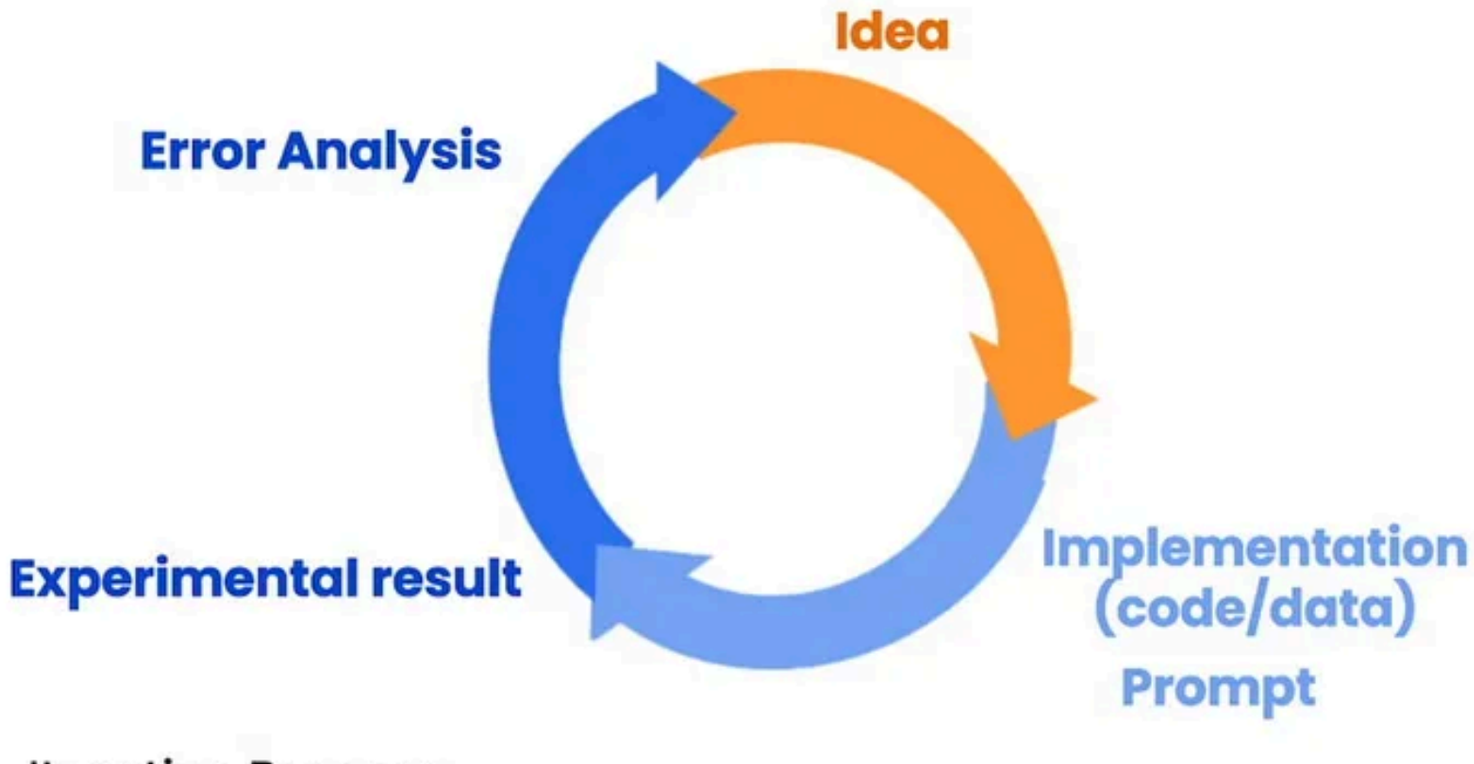
https://platform.openai.com/docs/guides/prompt-engineering

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

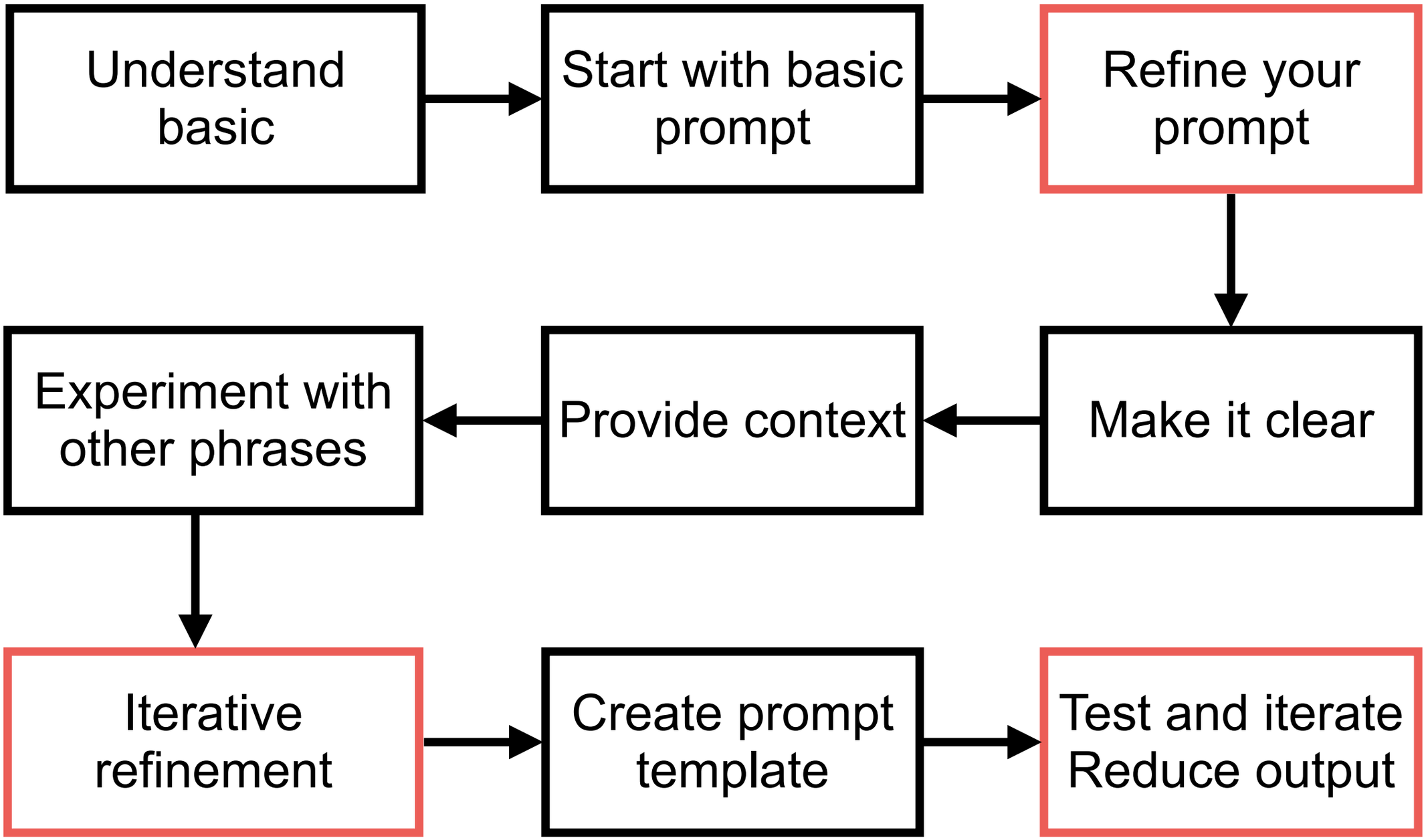


Iterative Process

- 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

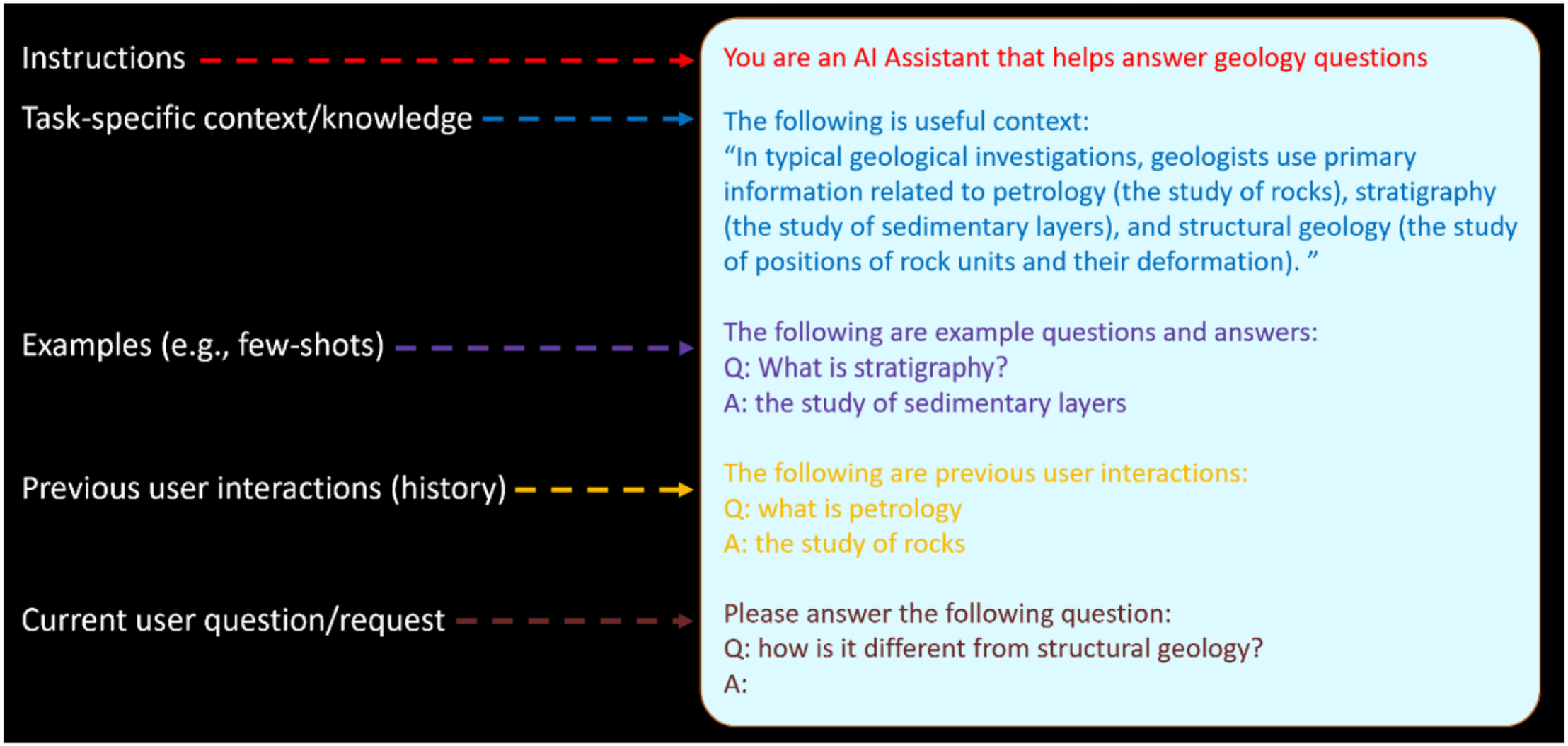


Better Prompt

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

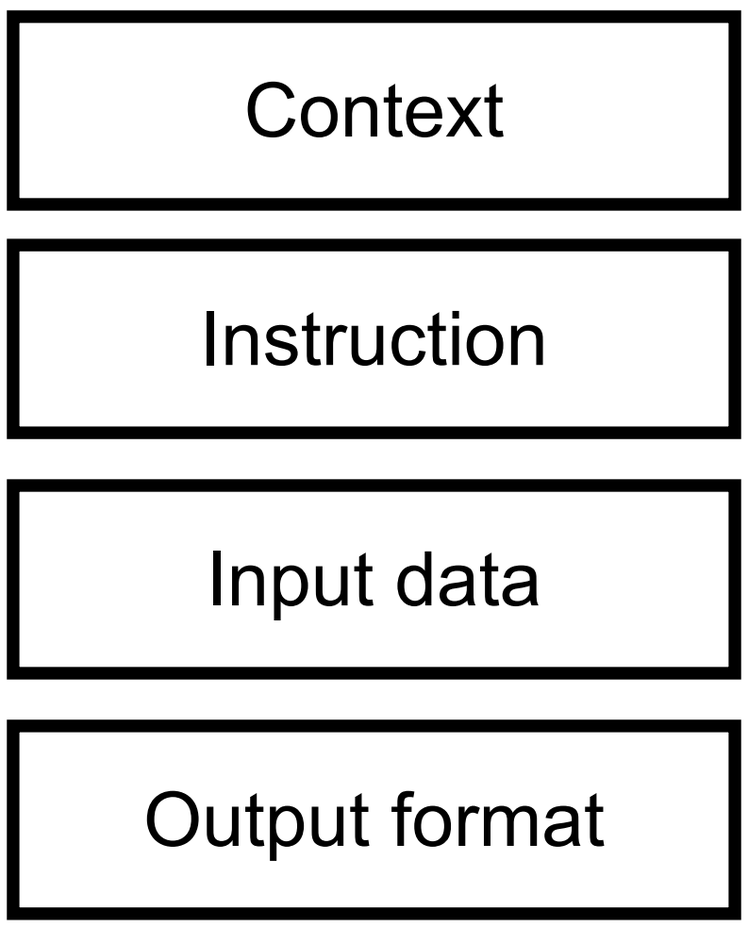
https://platform.openai.com/docs/guides/prompt-engineering/six-strategies-for-getting-better-results

Prompt Structure



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

Structure of Prompt



https://platform.openai.com/docs/guides/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 !!

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 Techniques

Zero-shot

Chain-of Thought (CoT)

Few-shot

Meta or structure

https://www.promptingguide.ai/techniques

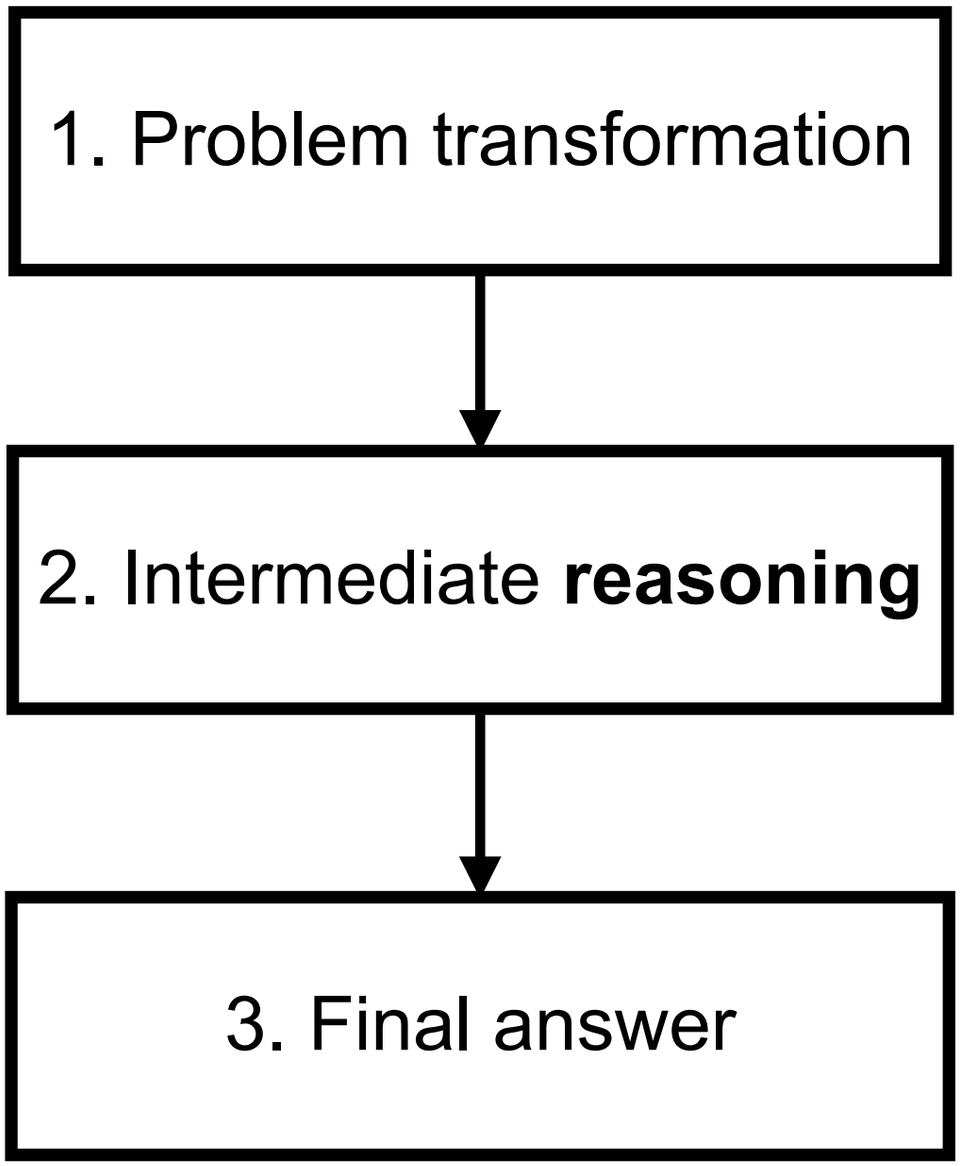
Chain of Thought Prompting (CoT)

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

Reasoning model !!

Chain of Thought Prompting (CoT)



eg. step-by-step

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)

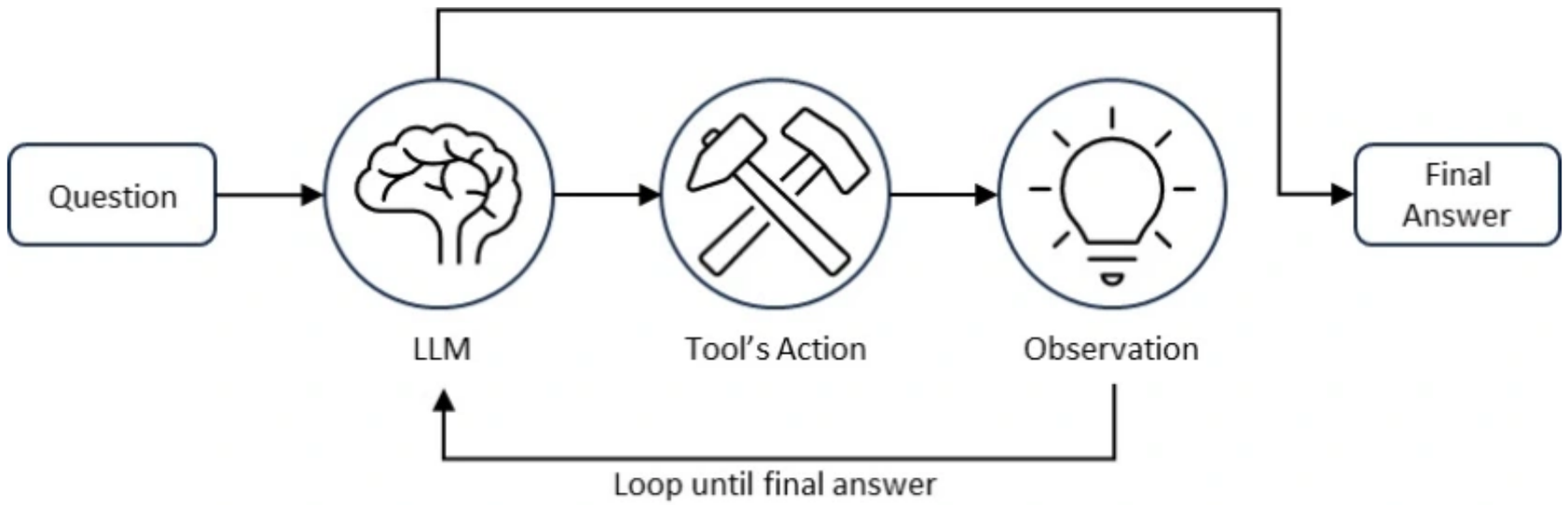
Q: A juggler can juggle 16 balls. Half of the balls are golf balls, and half of the golf balls are blue. How many blue golf balls are there?
A: ***Let's think step by step.***

(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

ReAct Prompt

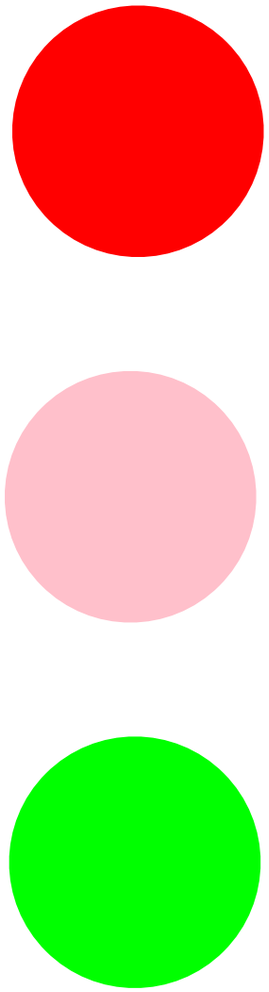
LLM reasoning and additional tools (expert)
Improve better answer



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

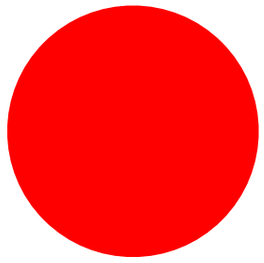
Workshop

RGB ?



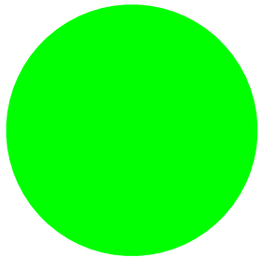
https://github.com/up1/workshop-ai-with-technical-team/tree/main/workshop/demo-rgb

RGB ?



[255, 0, 0]

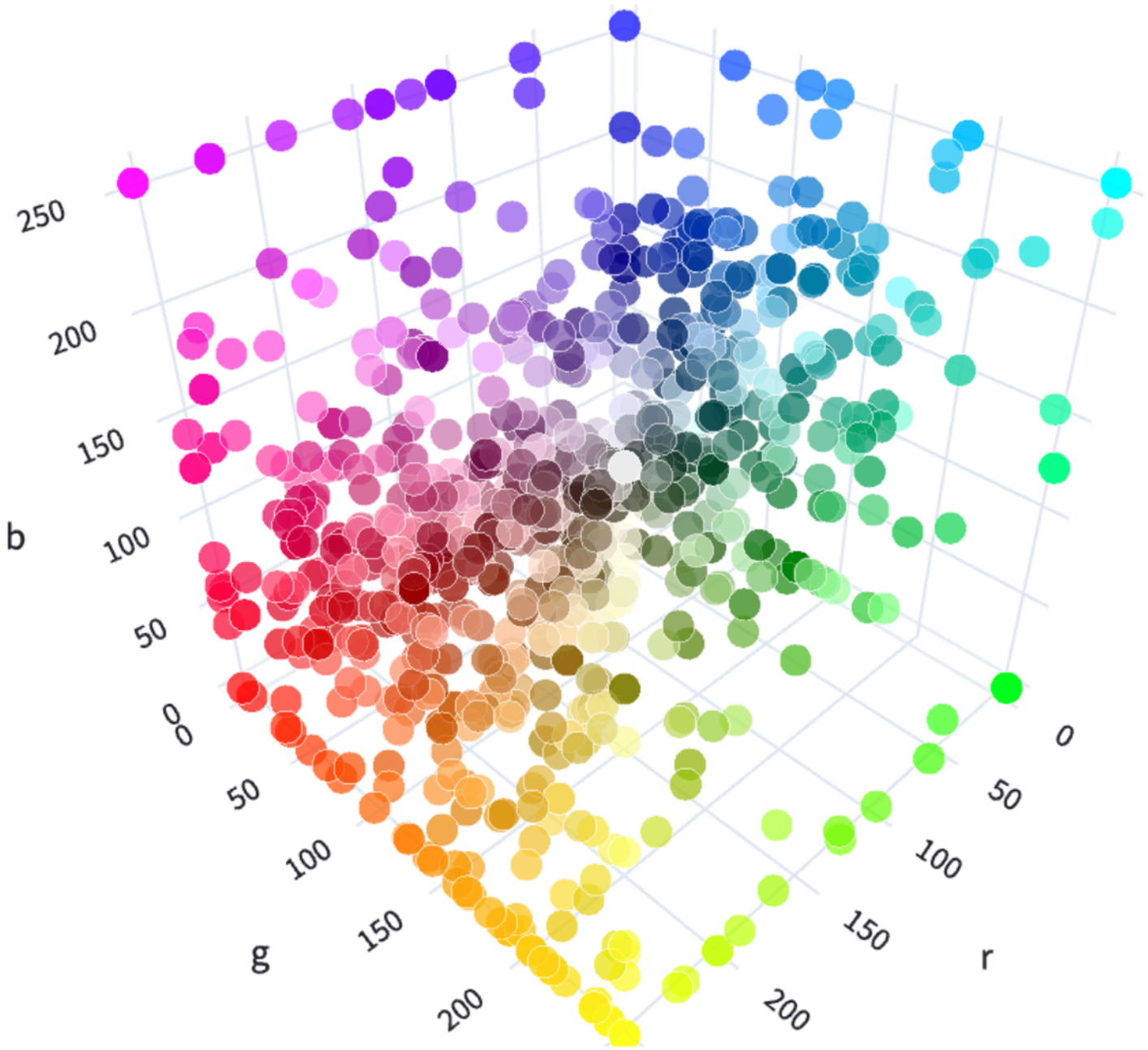
[255, 192, 203]



[0, 255, 0]

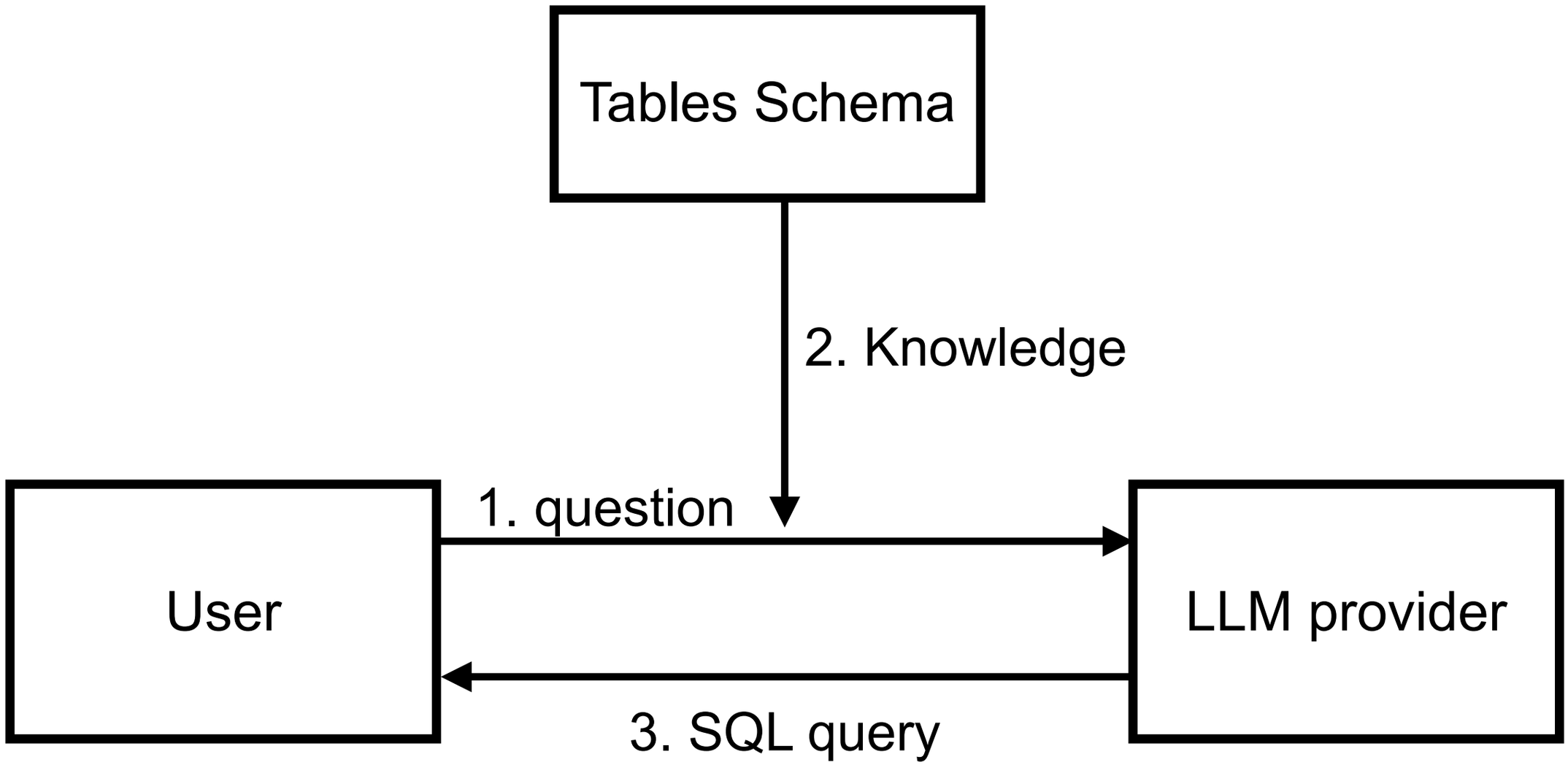
https://github.com/up1/workshop-ai-with-technical-team/tree/main/workshop/demo-rgb

Visualize RGB



https://huggingface.co/spaces/jphwang/colorful_vectors

Text-to-SQL



https://github.com/up1/workshop-ai-with-technical-team/tree/main/workshop/demo-sql

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

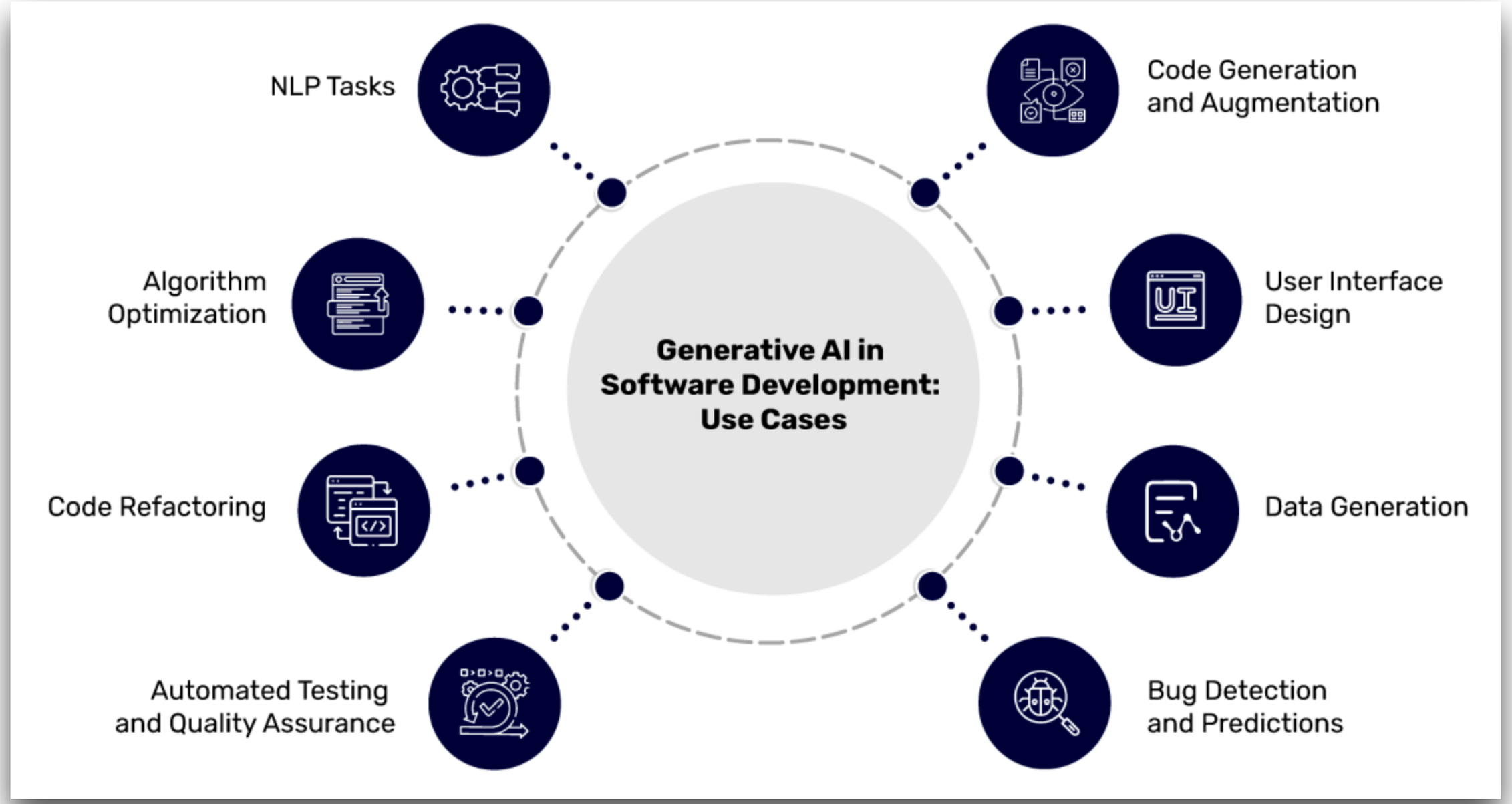
- FF & experimentation
- Logging
- Monitoring & alerting
- Staged rollouts

Maintenance

- Debug production
- Documentation
- Runbook examples
- Mitigation runbook

pragmaticengineer.com

SDLC



Impacts with productivity ?

Automated simple tasks

Improve quality and reliability

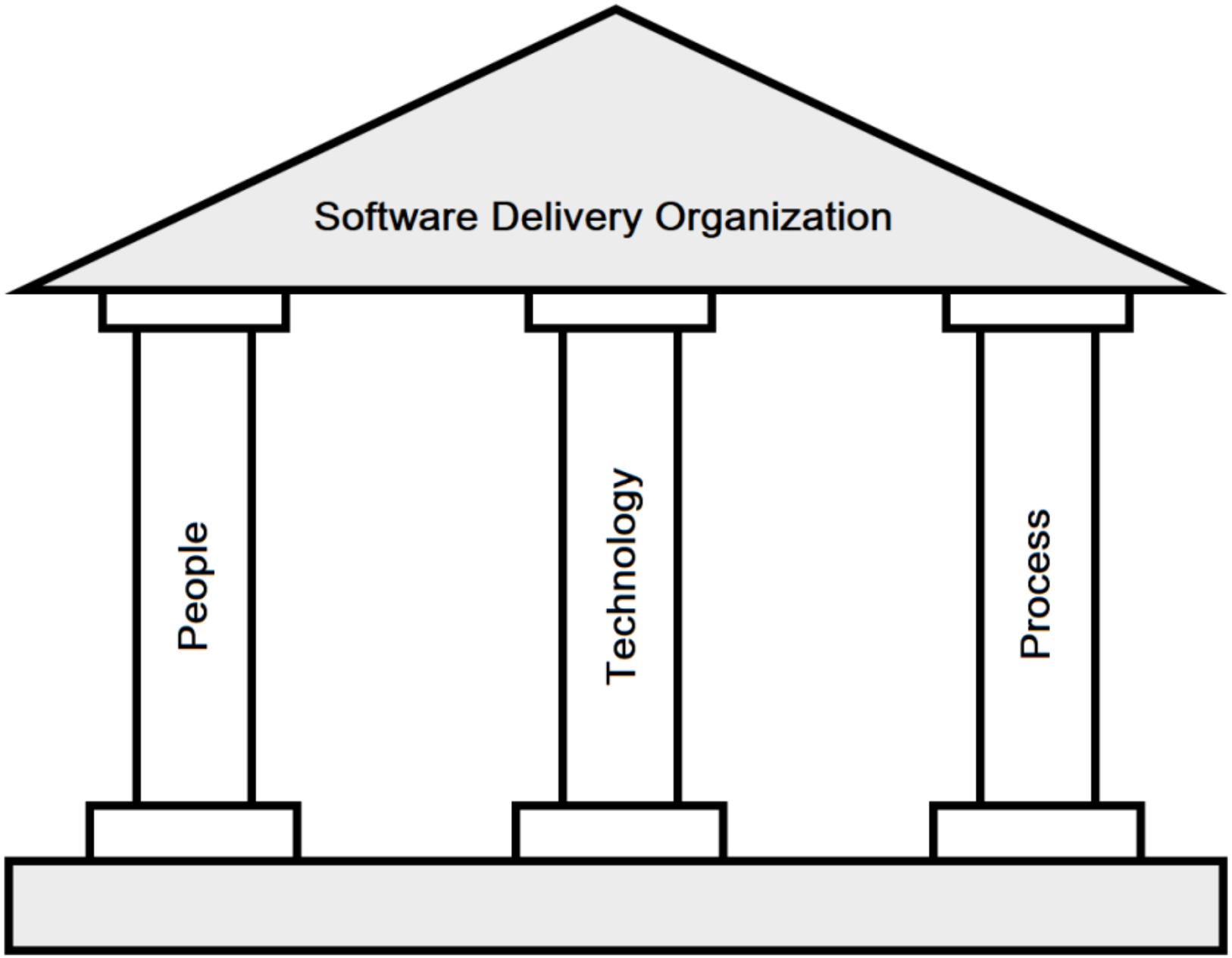
Improve communication

Faster prototype

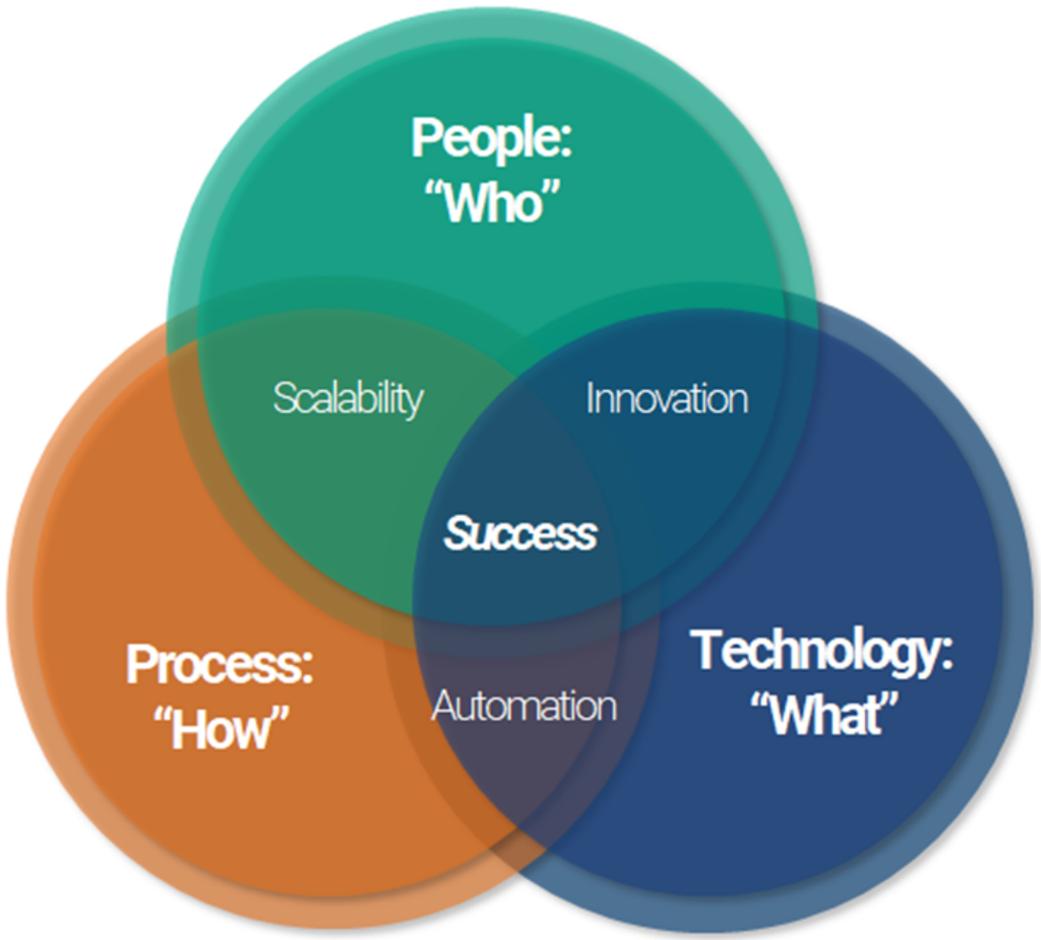
Generative AI isn't just a tool it's your **team member**

Trust, but verify output !!

3 Pillar of Software Development



3 Pillar of Software Development

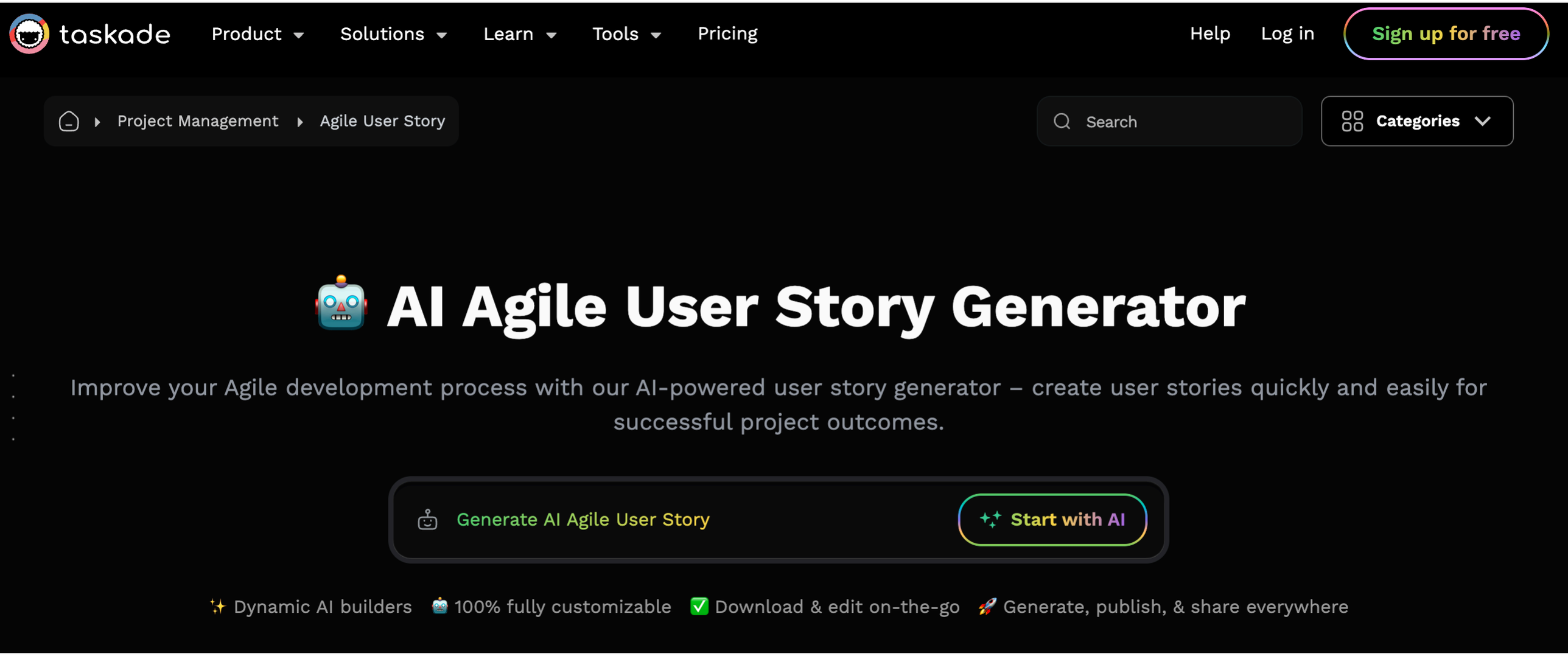


Requirement and Analysis

Requirement and Analysis

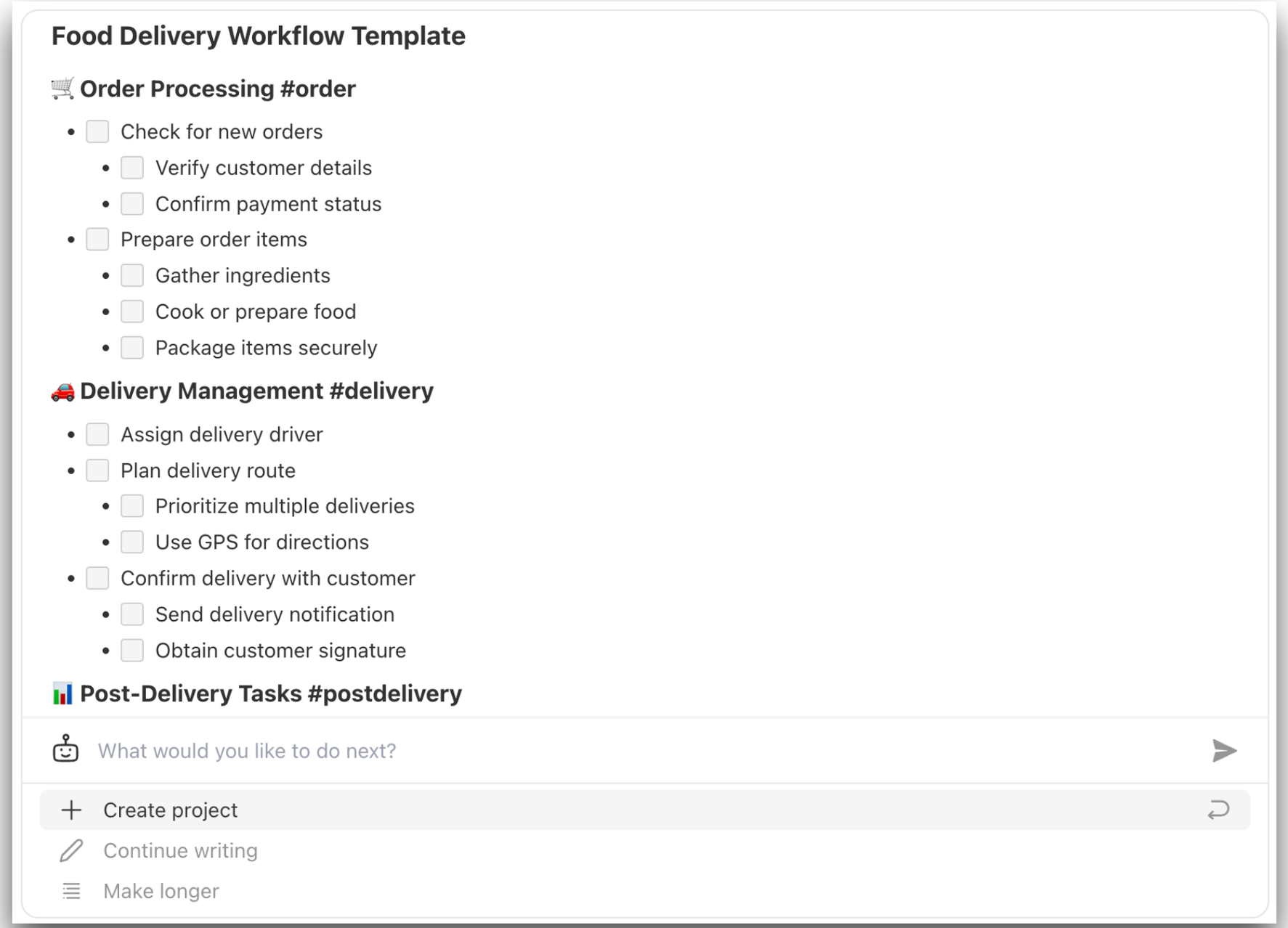
Requirements writing and analysis
User story generation

Agents AI for Automated tasks



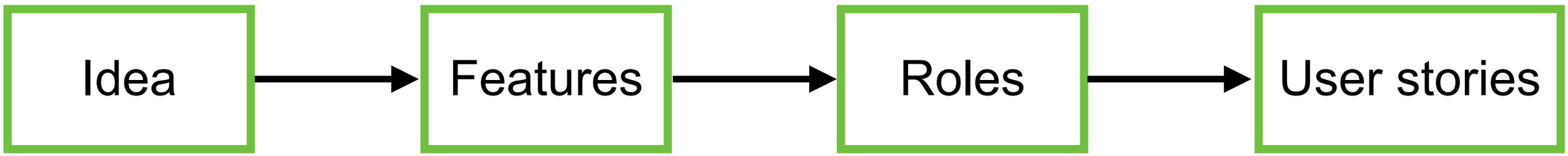
https://www.taskade.com/generate/project-management/agile-user-story

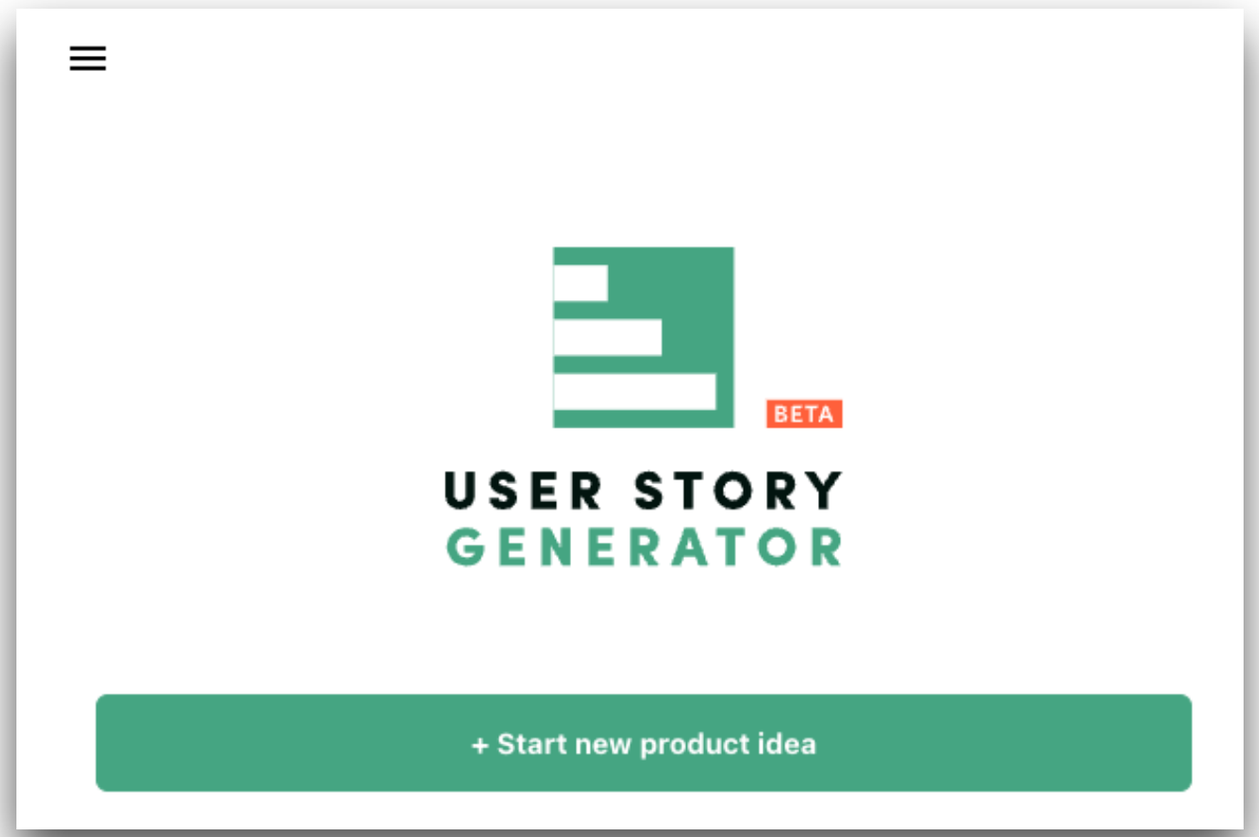
Example with food delivery



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

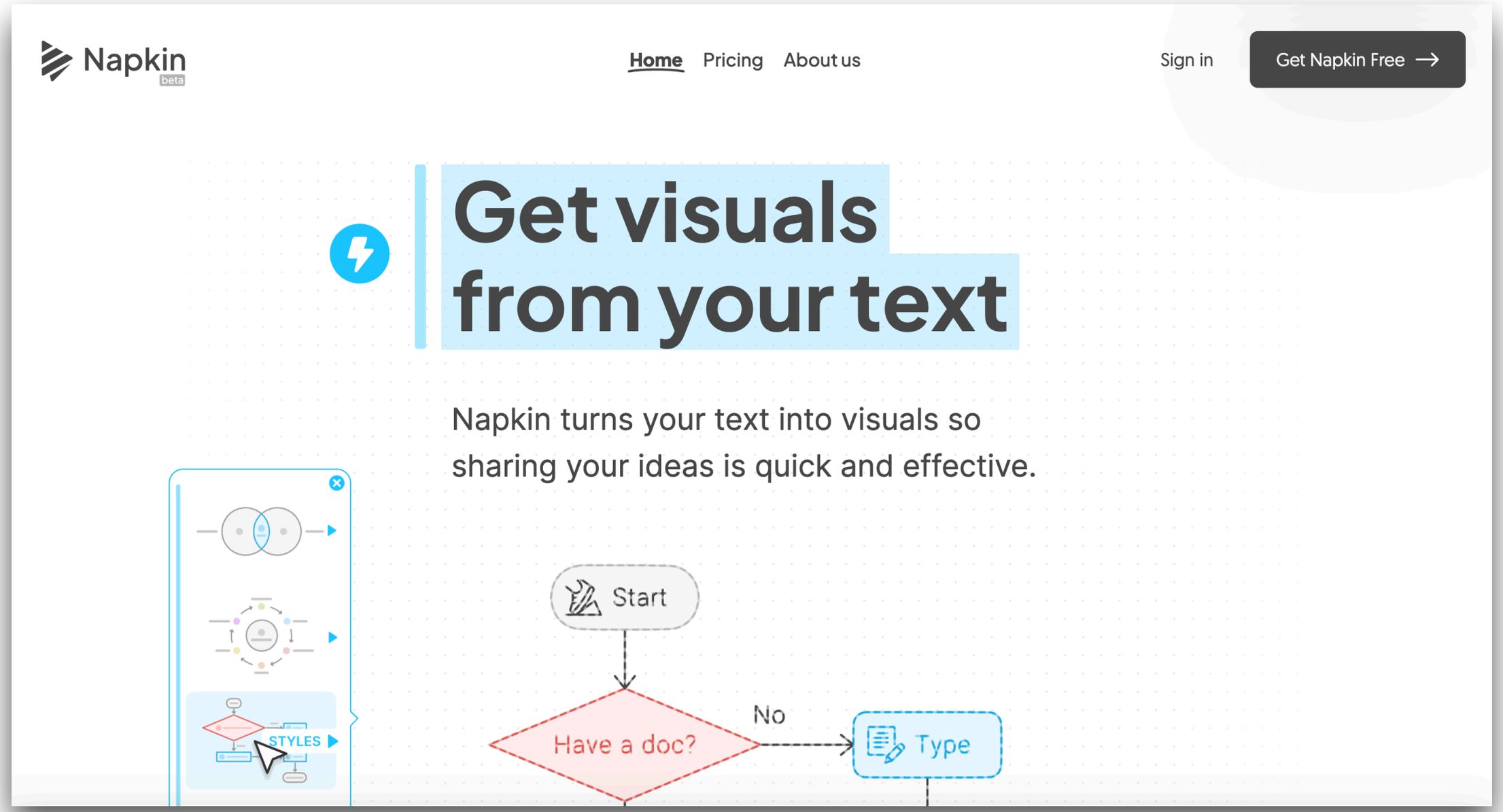
User Story Generator





https://userstorygenerator.ai/

Napkin



https://www.napkin.ai/

Requirement analysis

Clarify of User requirement ?

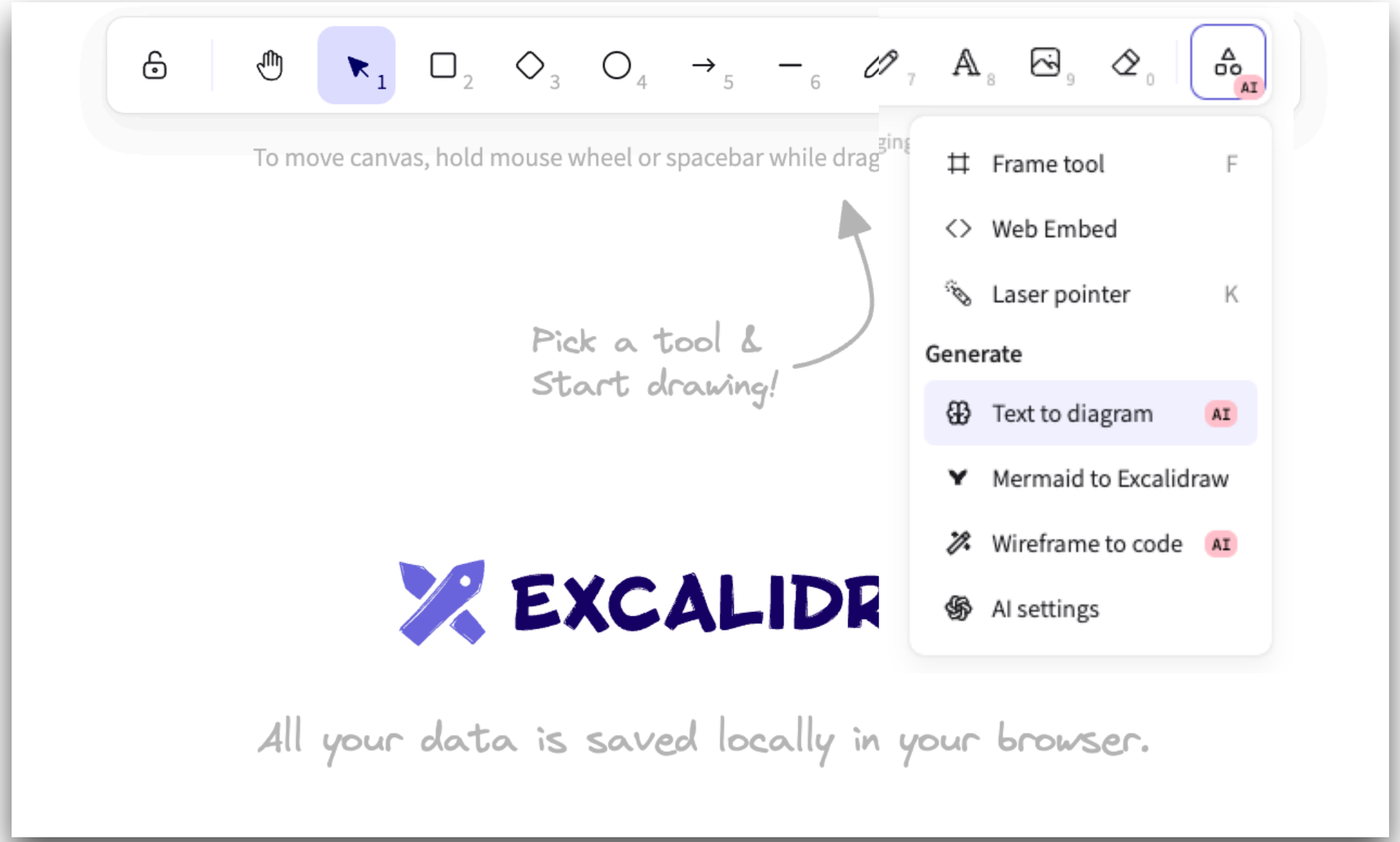
https://github.com/up1/workshop-ai-with-technical-team/wiki/Requirement-analysis

Design Process

Design

Architecture writing assistance
Sequence flow diagram generation
Data modeling
UX/UI design assistance

Excalidraw with AI



https://excalidraw.com/

Demo

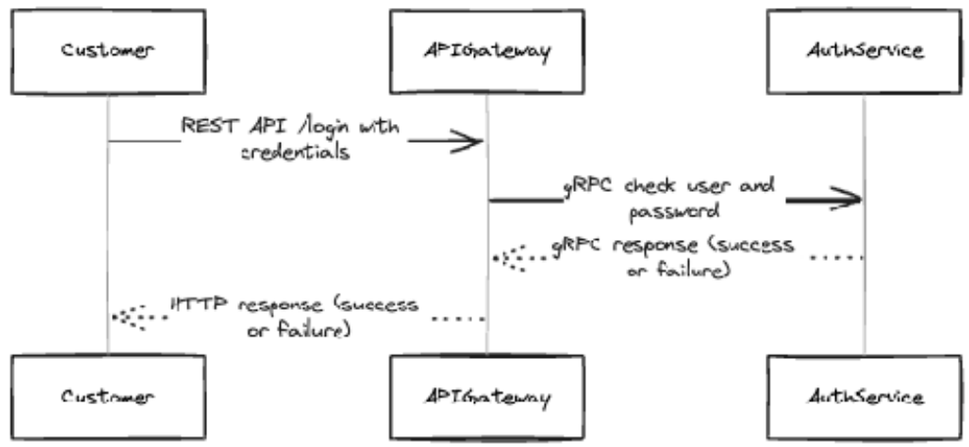
Text to diagram AI Beta | Mermaid

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
```

Preview

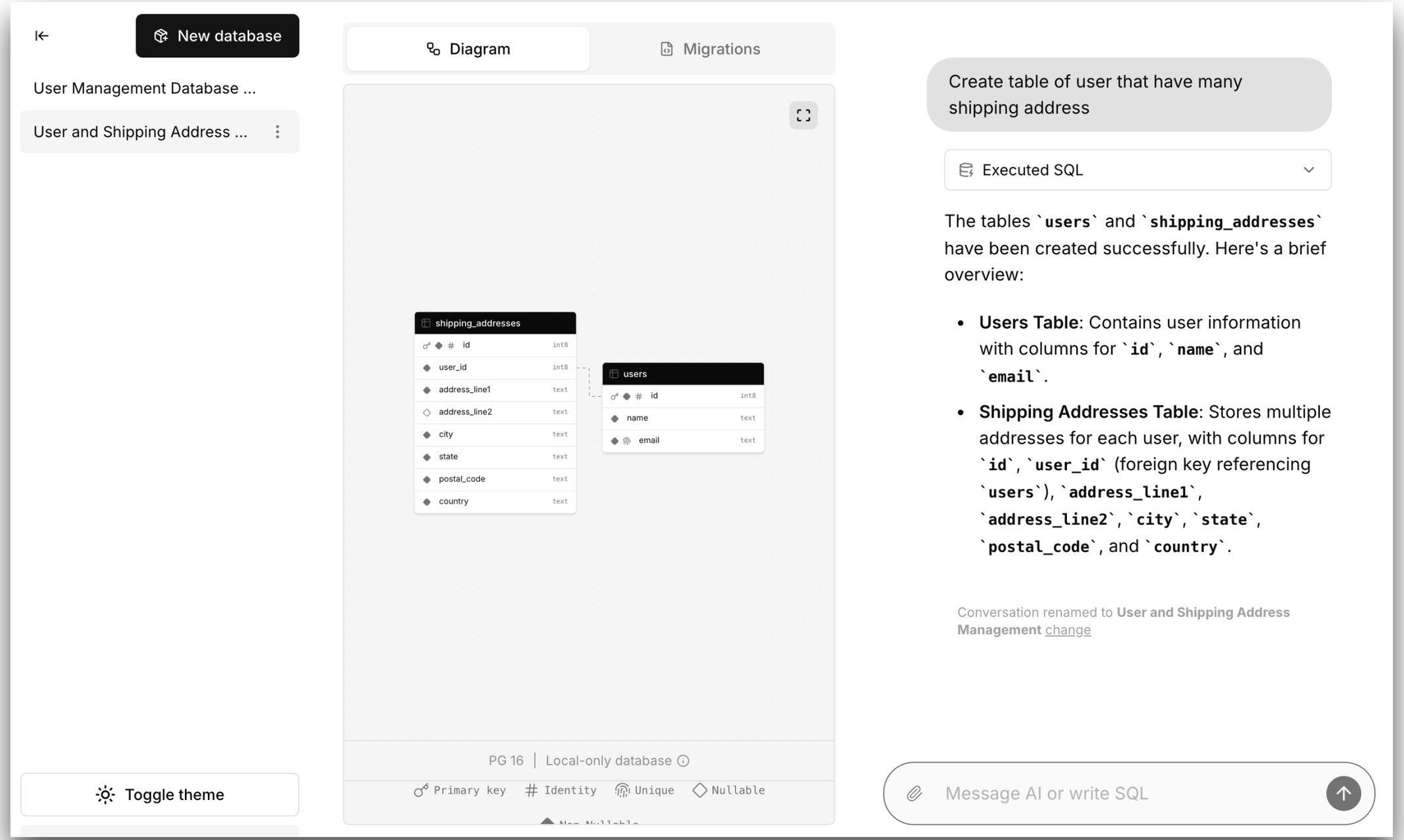


Generate → Cmd Enter

View as Mermaid →

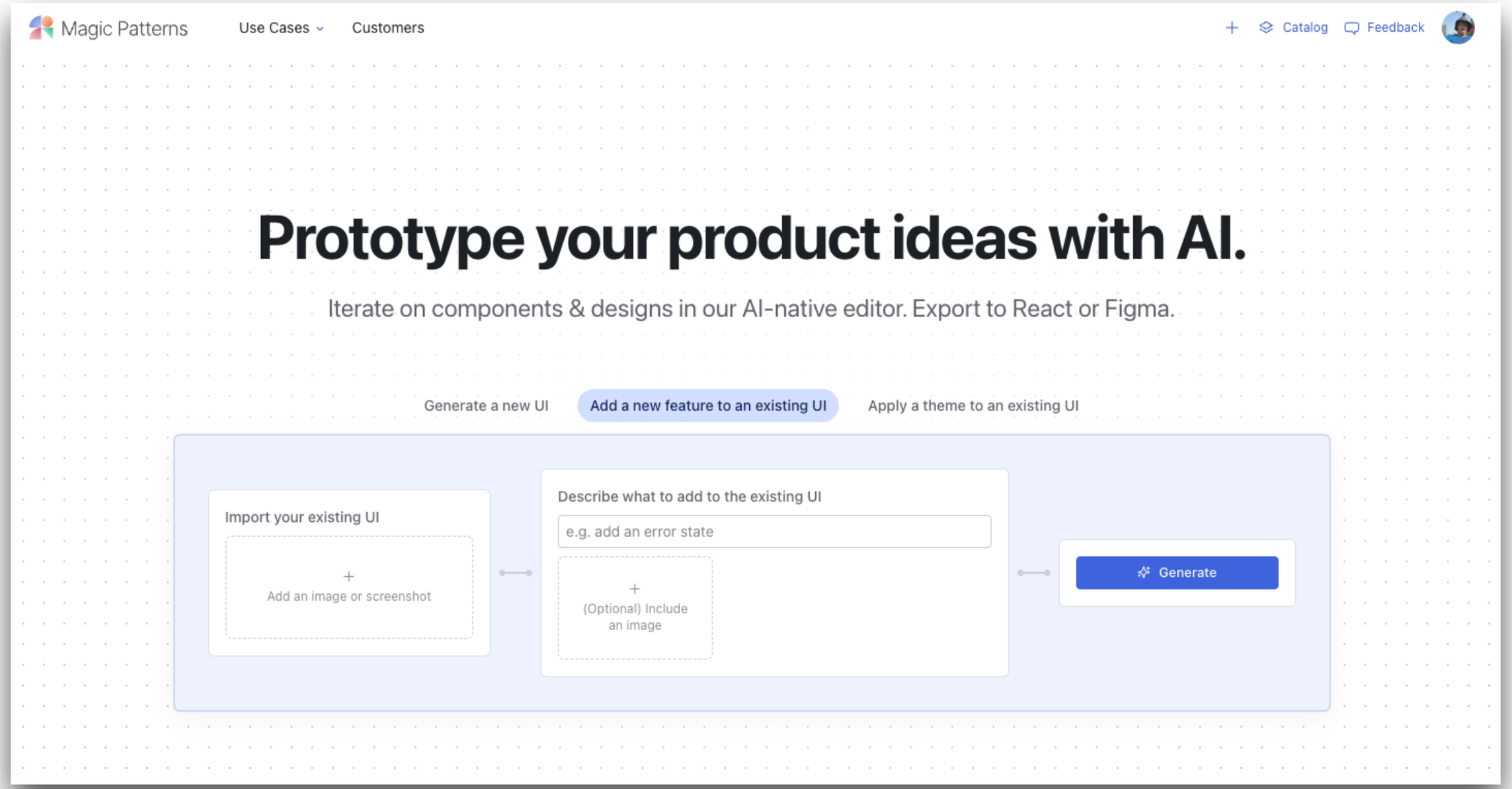
Insert →

Database



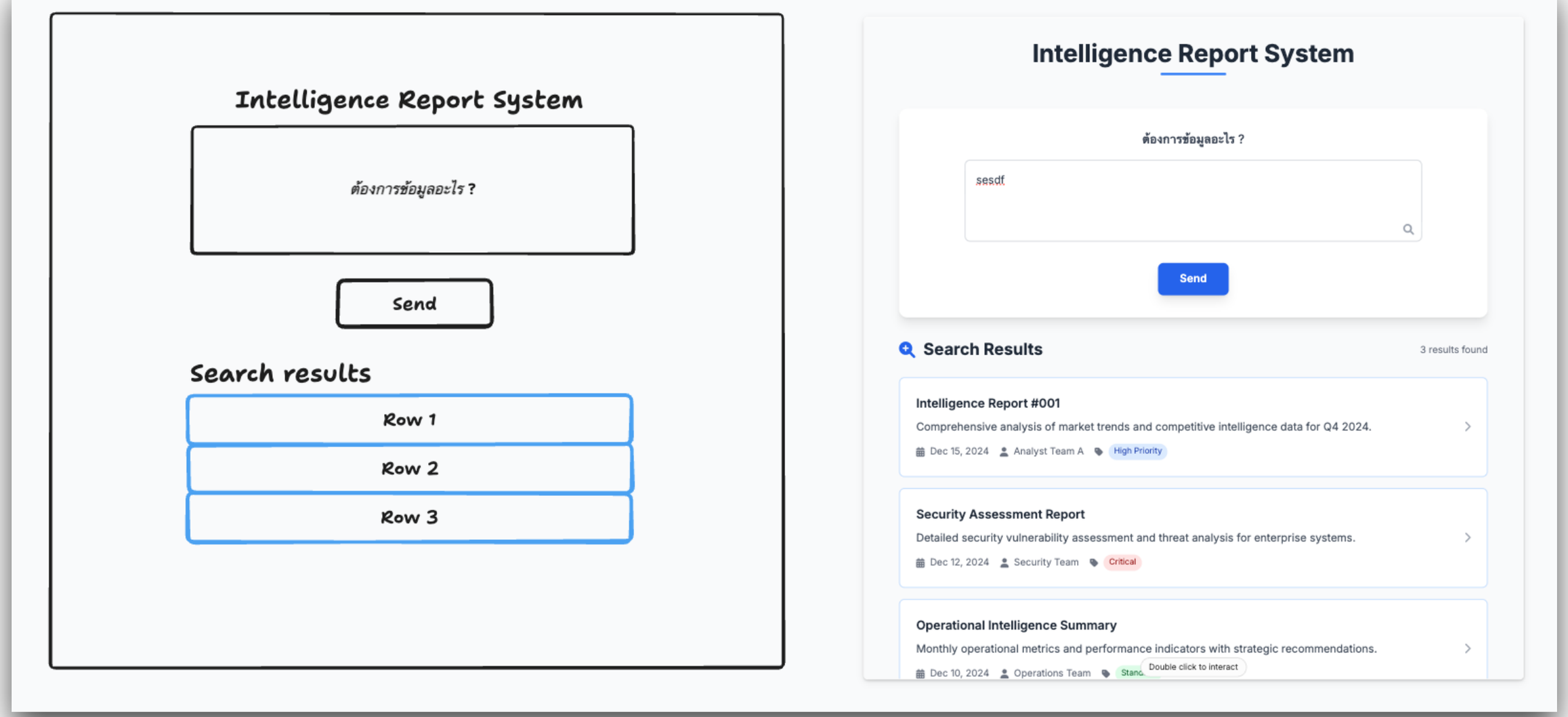
https://database.build/

Magic Pattern



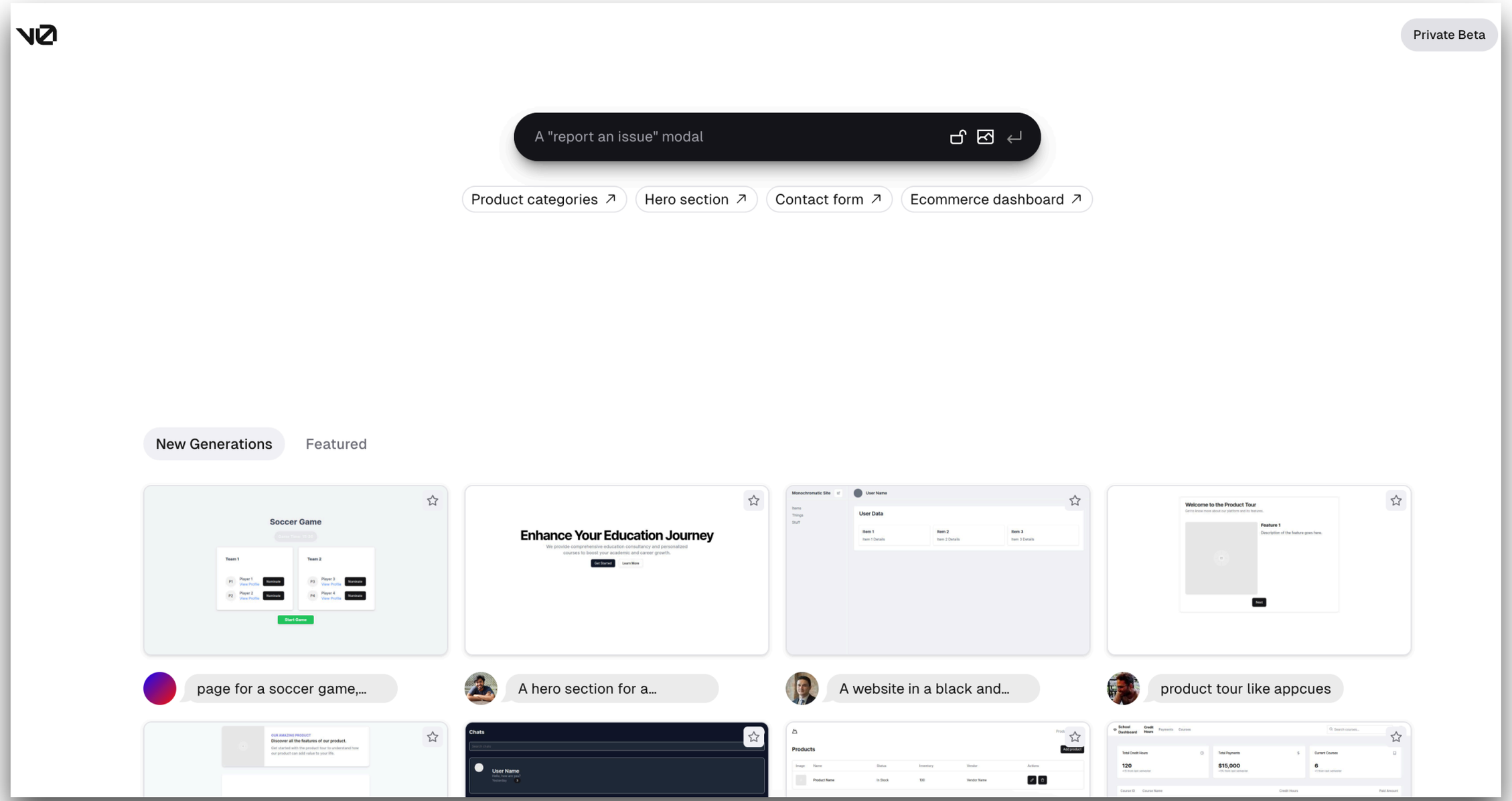
https://www.magicpatterns.com/

Make Real



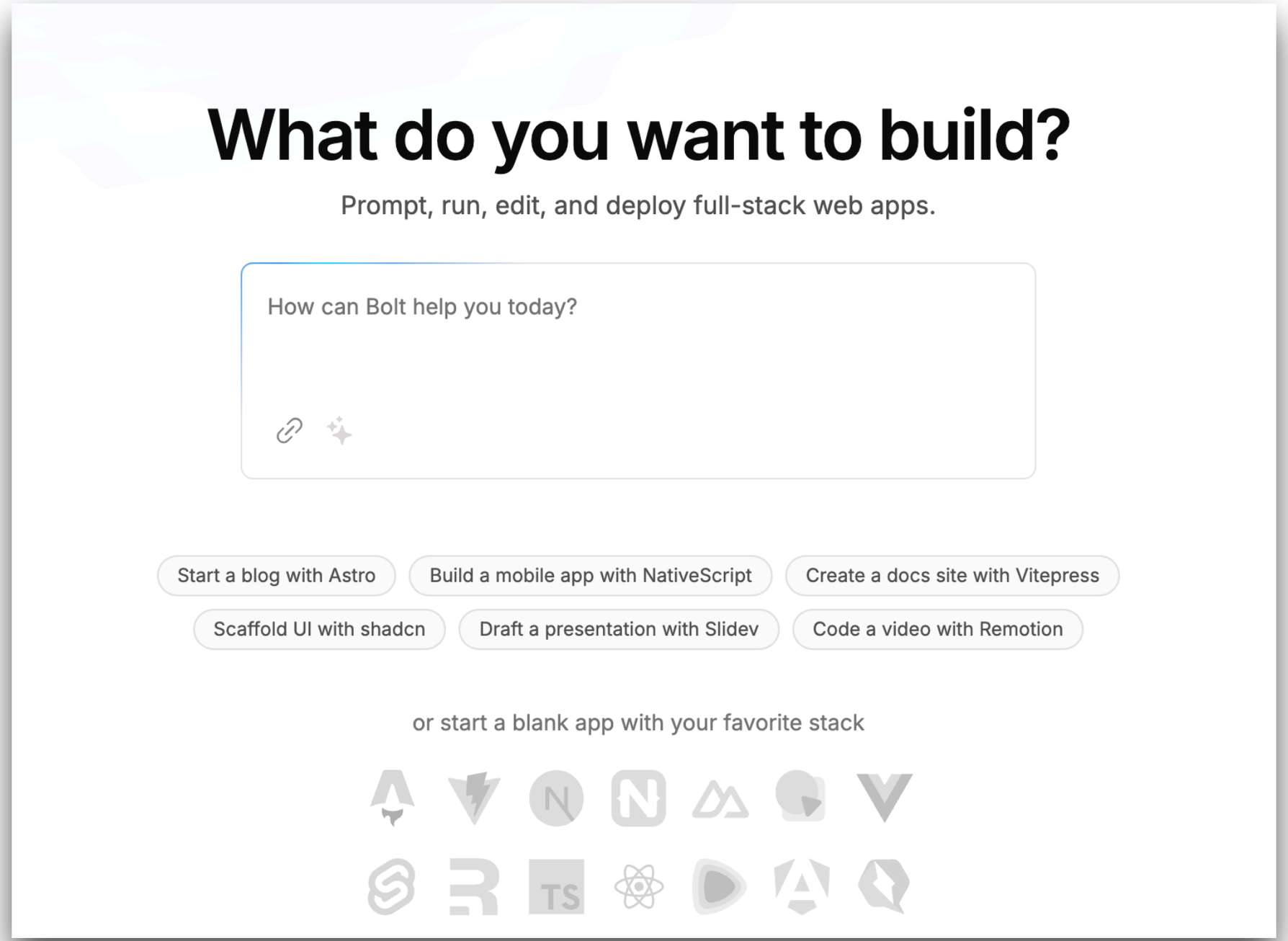
https://github.com/tldraw/make-real

V0.dev



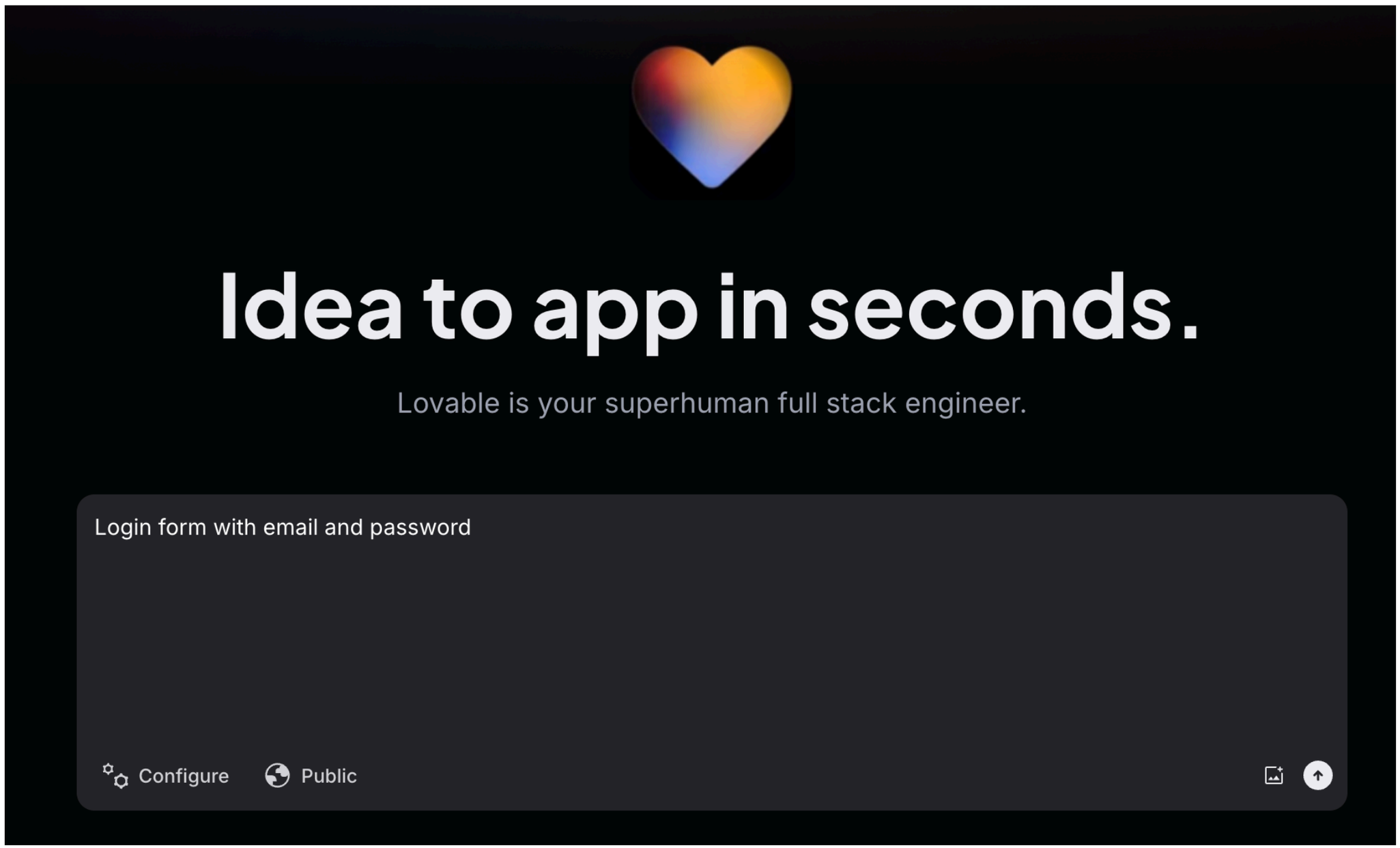
https://v0.dev/

Bolt.new



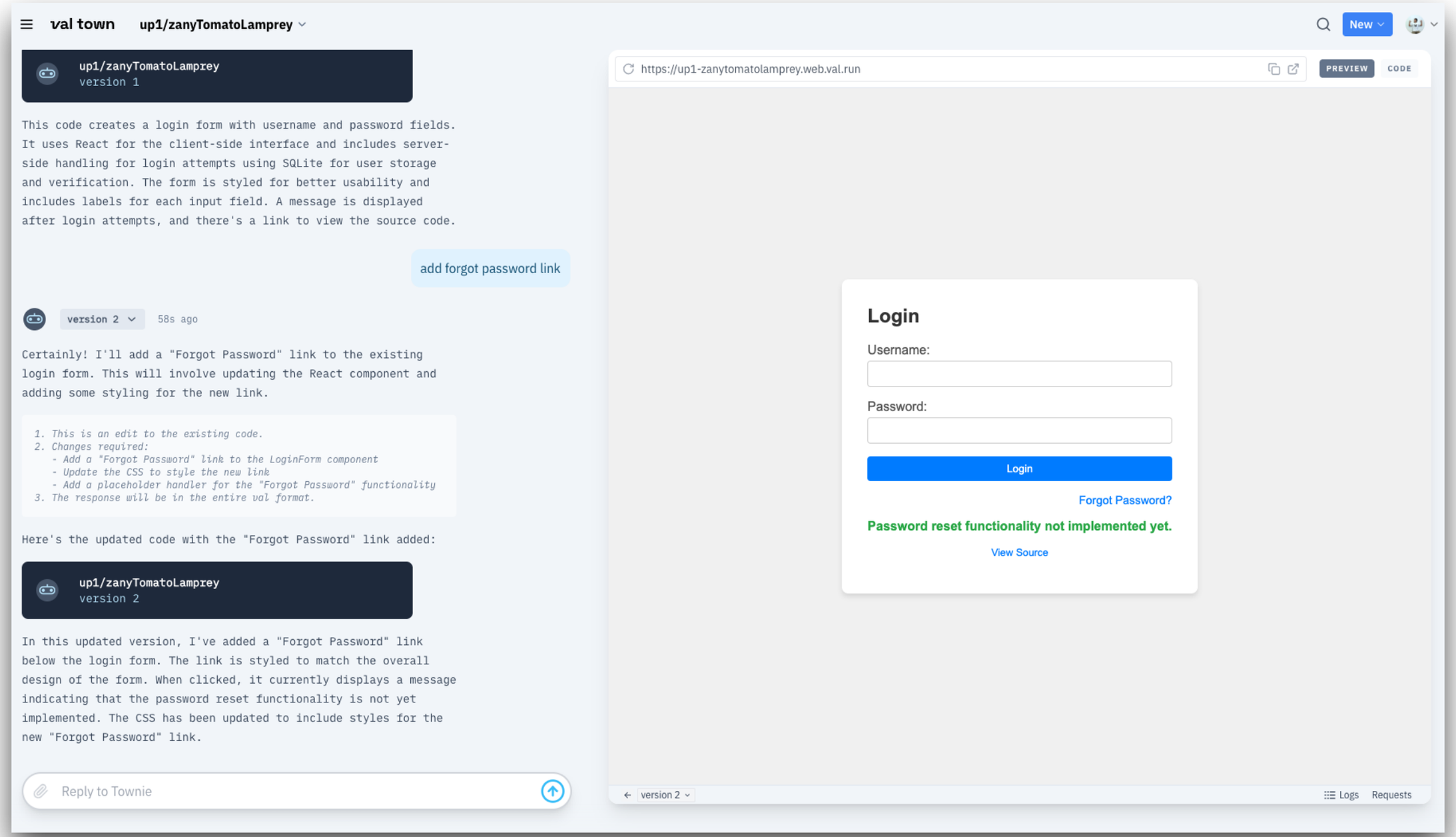
https://bolt.new/

Lovable



https://lovable.dev/

Val Town



https://www.val.town/

Google Stitch

https://stitch.withgoogle.com/

Development Process

Develop

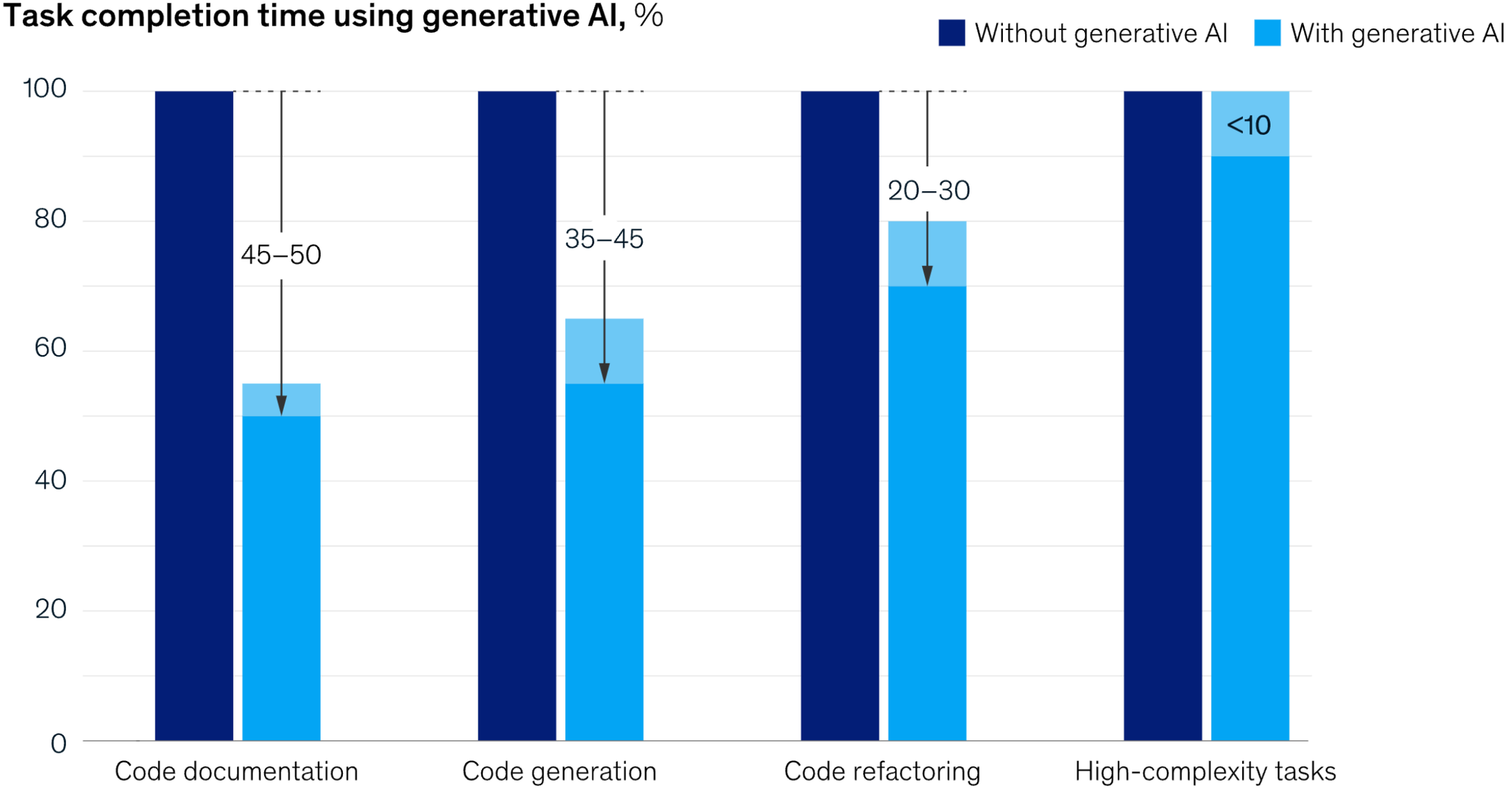
Requirement | Design | **Develop** | Testing | Deploy

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.

Task completion time using generative AI, %

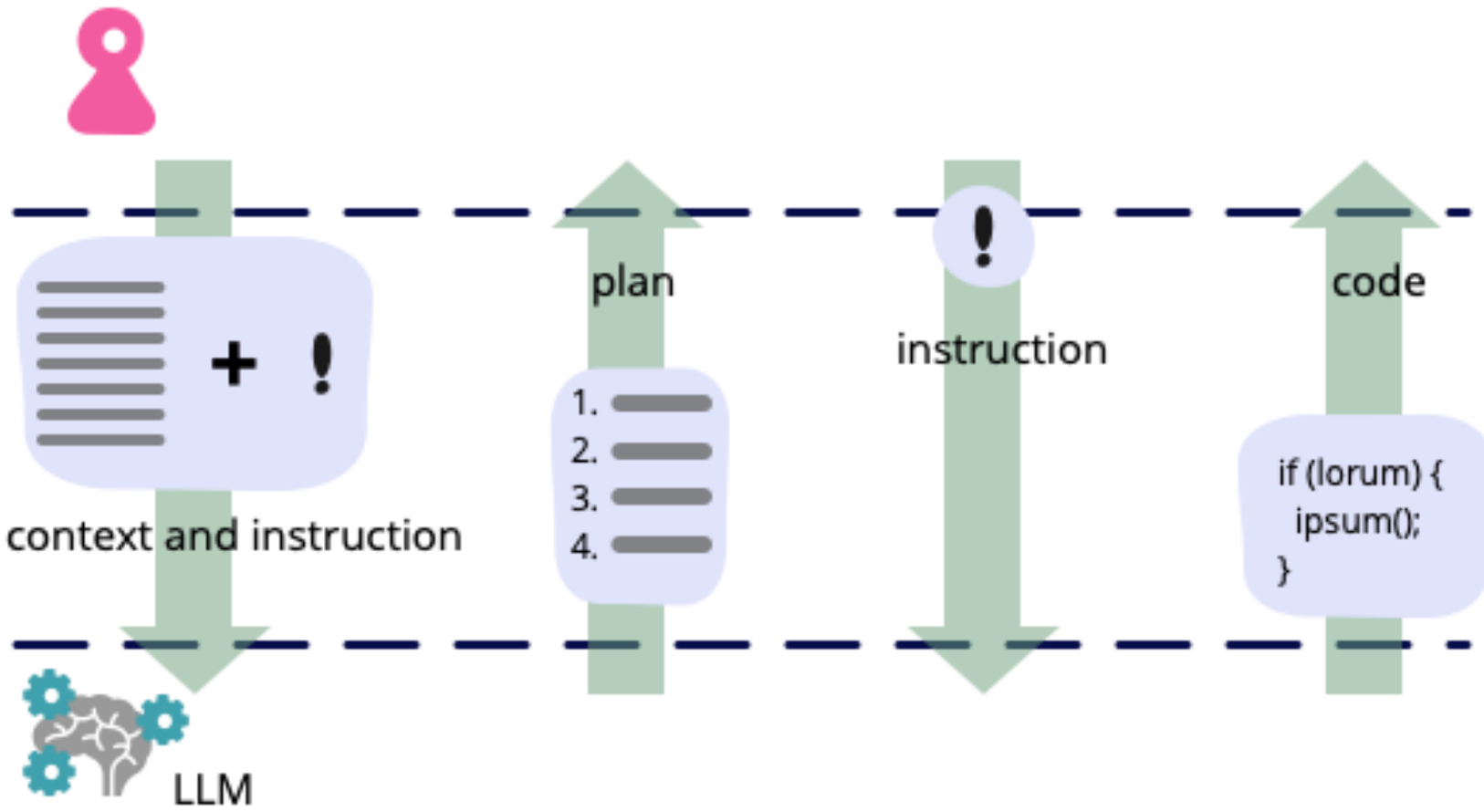


https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/unleashing-developer-productivity-with-generative-ai

Category of Tools

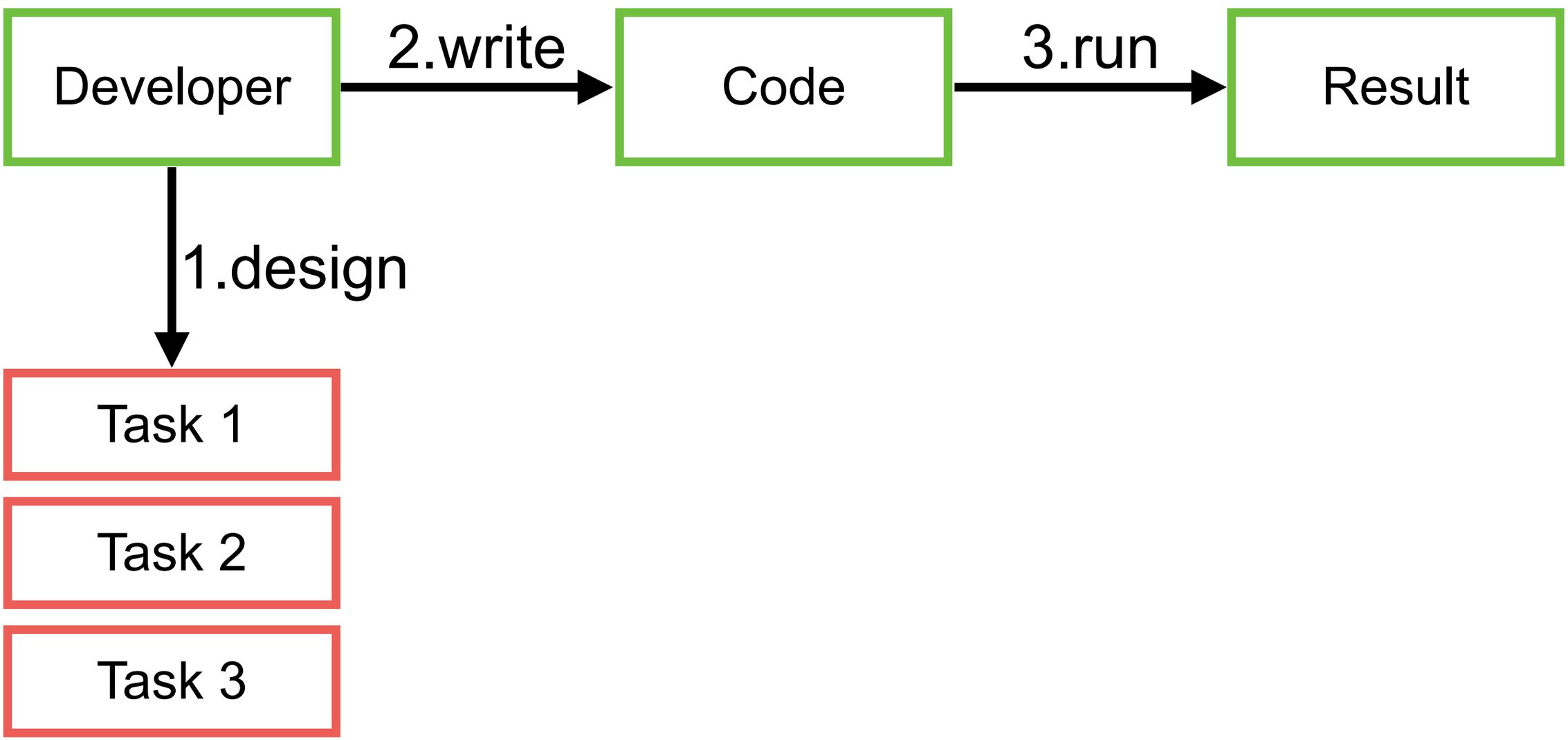
Chat AI	Code AI	AI Agent Public and Local
ChatGPT	Copilot in VSCode	Aider
Gemini	Cursor IDE	Claude code
Claude.ai	Windsurf	Gemini CLI
DeepSeek	Zed	Codex CLI
		Qwen3-coder

Development

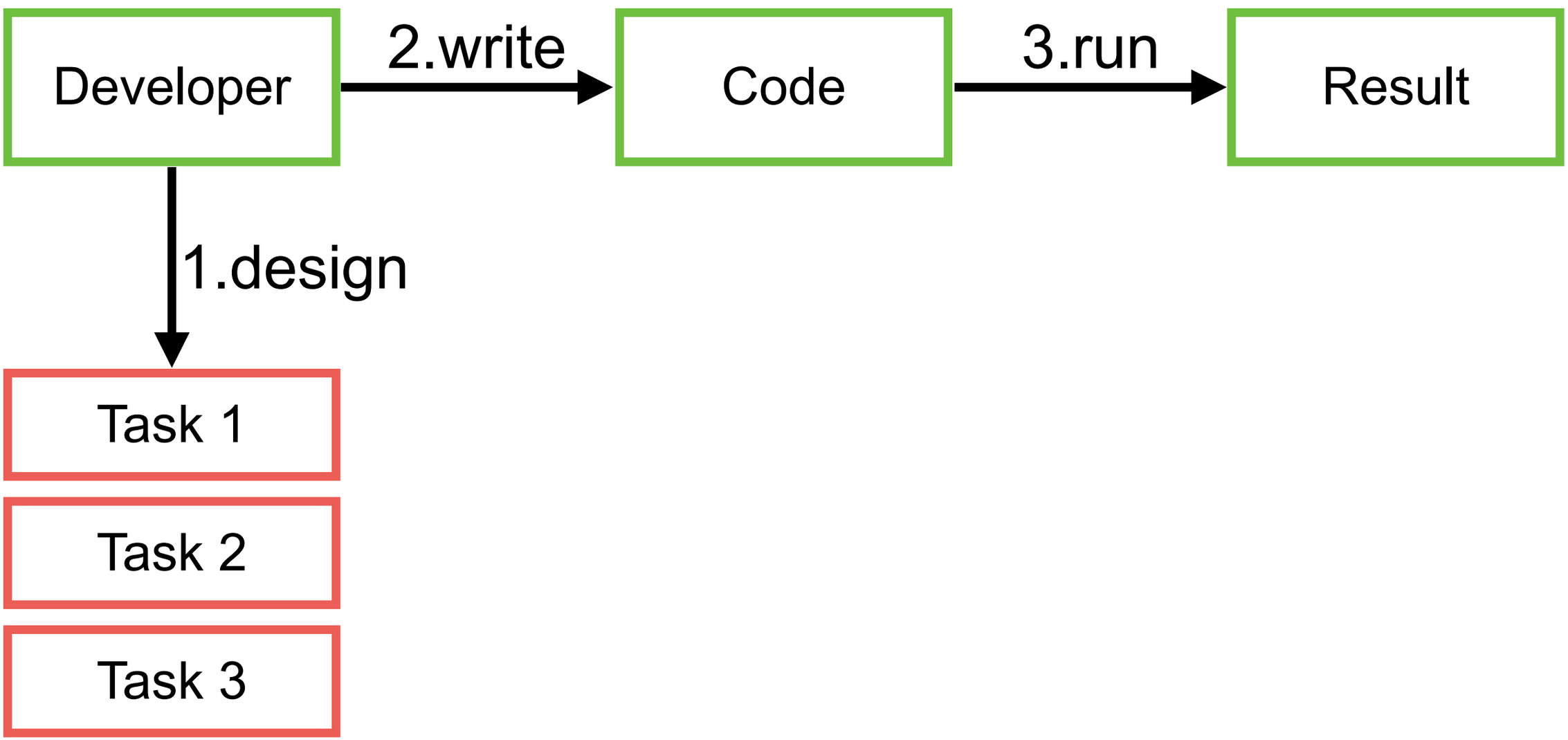


https://martinfowler.com/articles/2023-chatgpt-xu-hao.html

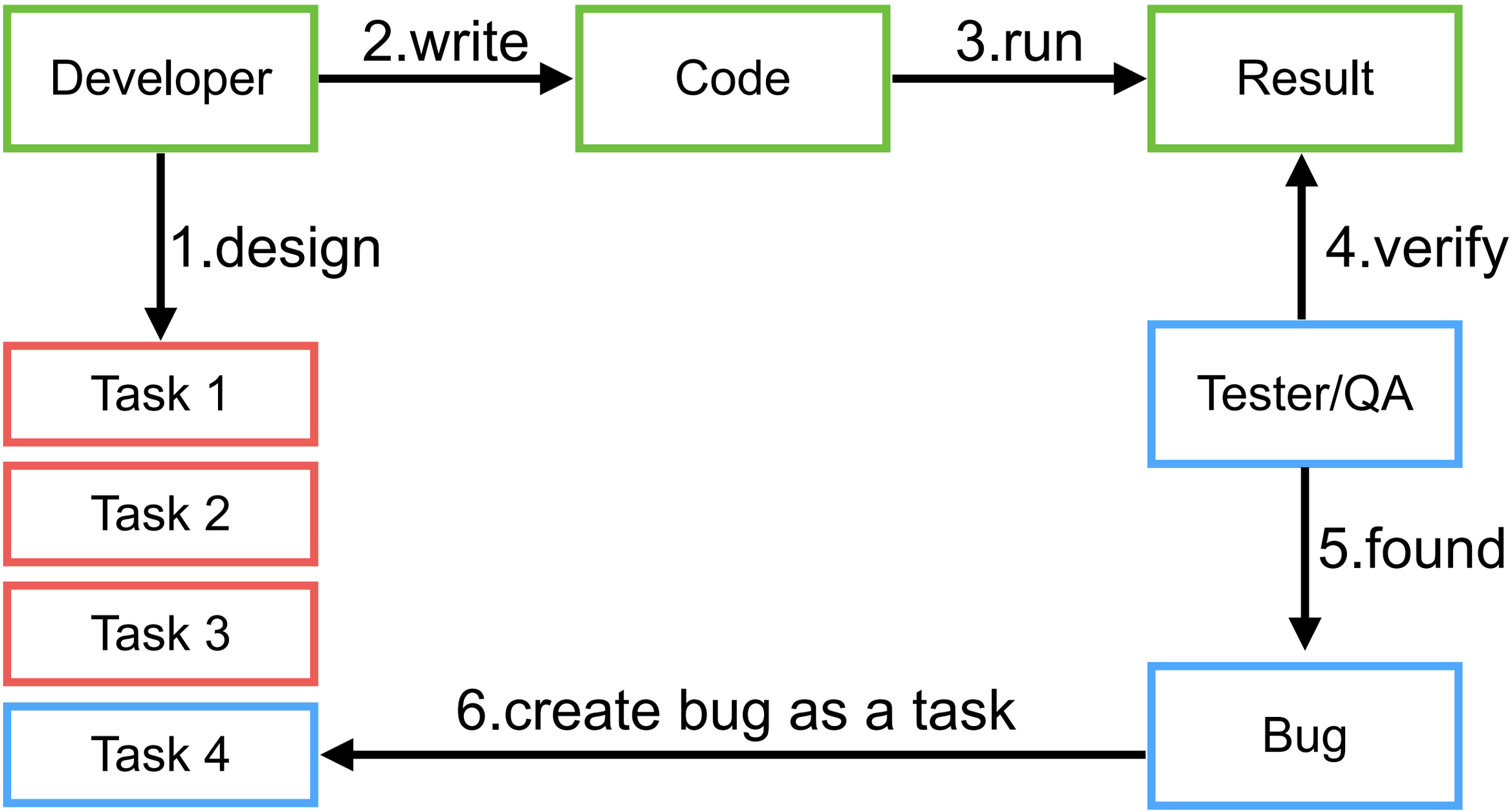
Development



Development



Development + Testing



Main Features

Ask question

Generate code

Refactor code

Document code

Find problems in your code

Pair programming with AI

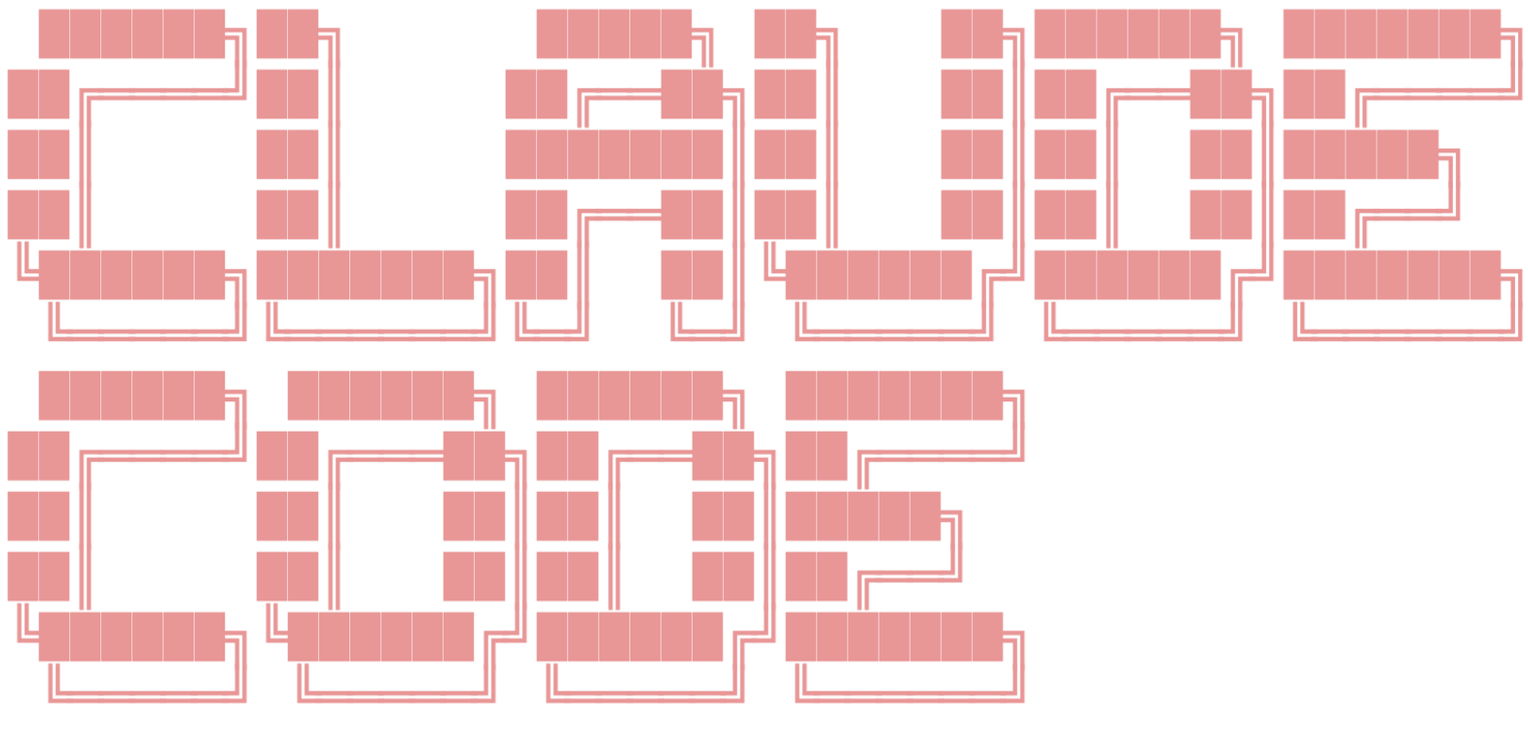
AI Pair Programming

Aider is AI pair programming in your terminal

Aider lets you pair program with LLMs, to edit code in your local git repository. Start a new project or work with an existing git repo. Aider works best with GPT-4o & Claude 3.5 Sonnet and can connect to almost any LLM.

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

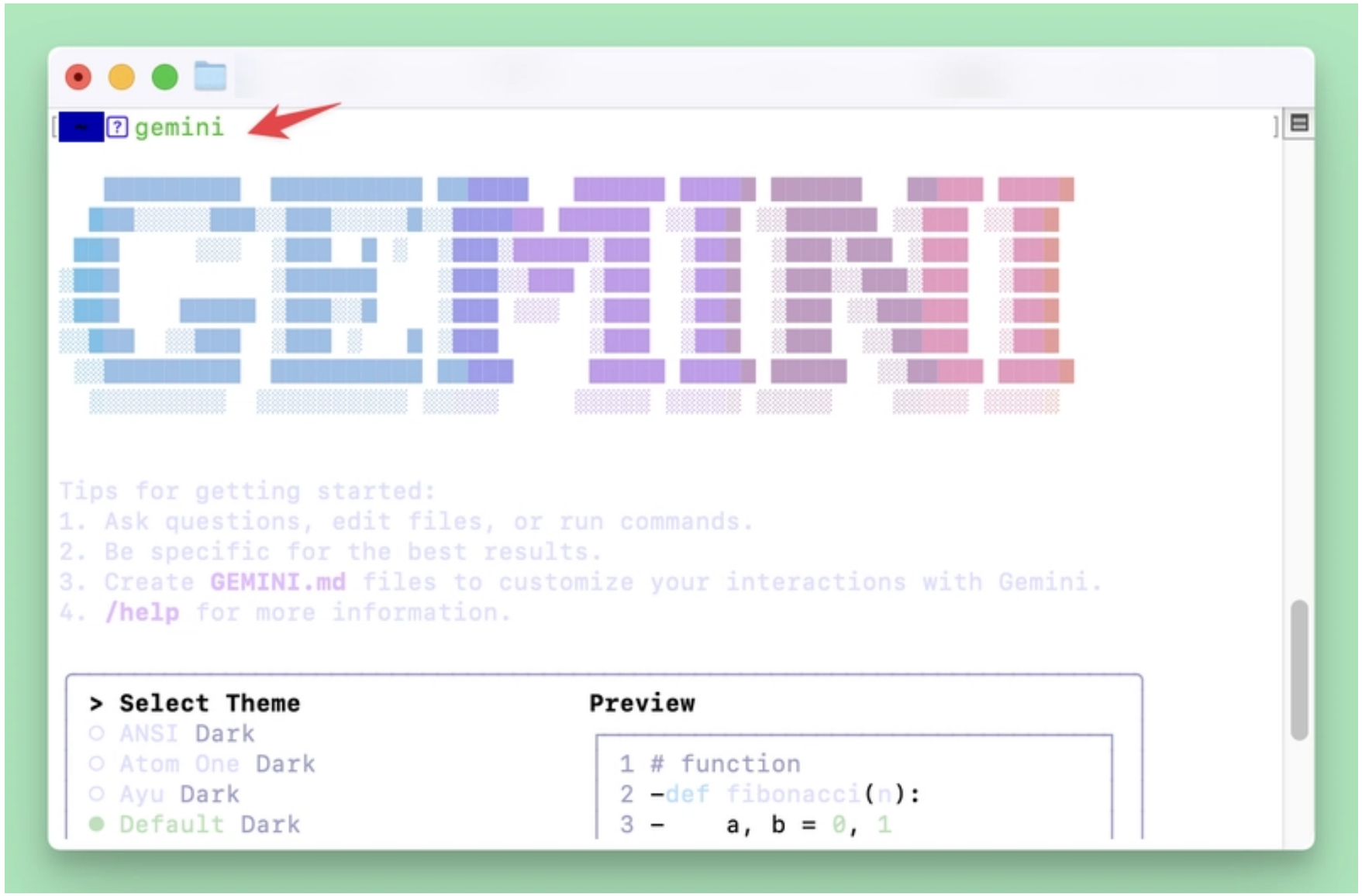
```
* Welcome to Claude Code research preview!
```



```
Claude Code is billed based on API usage through your Anthropic Console account.

Pricing may evolve as we move towards general availability.

Press Enter to login to your Anthropic Console account…
```

2025/02

https://www.anthropic.com/news/claude-3-7-sonnet

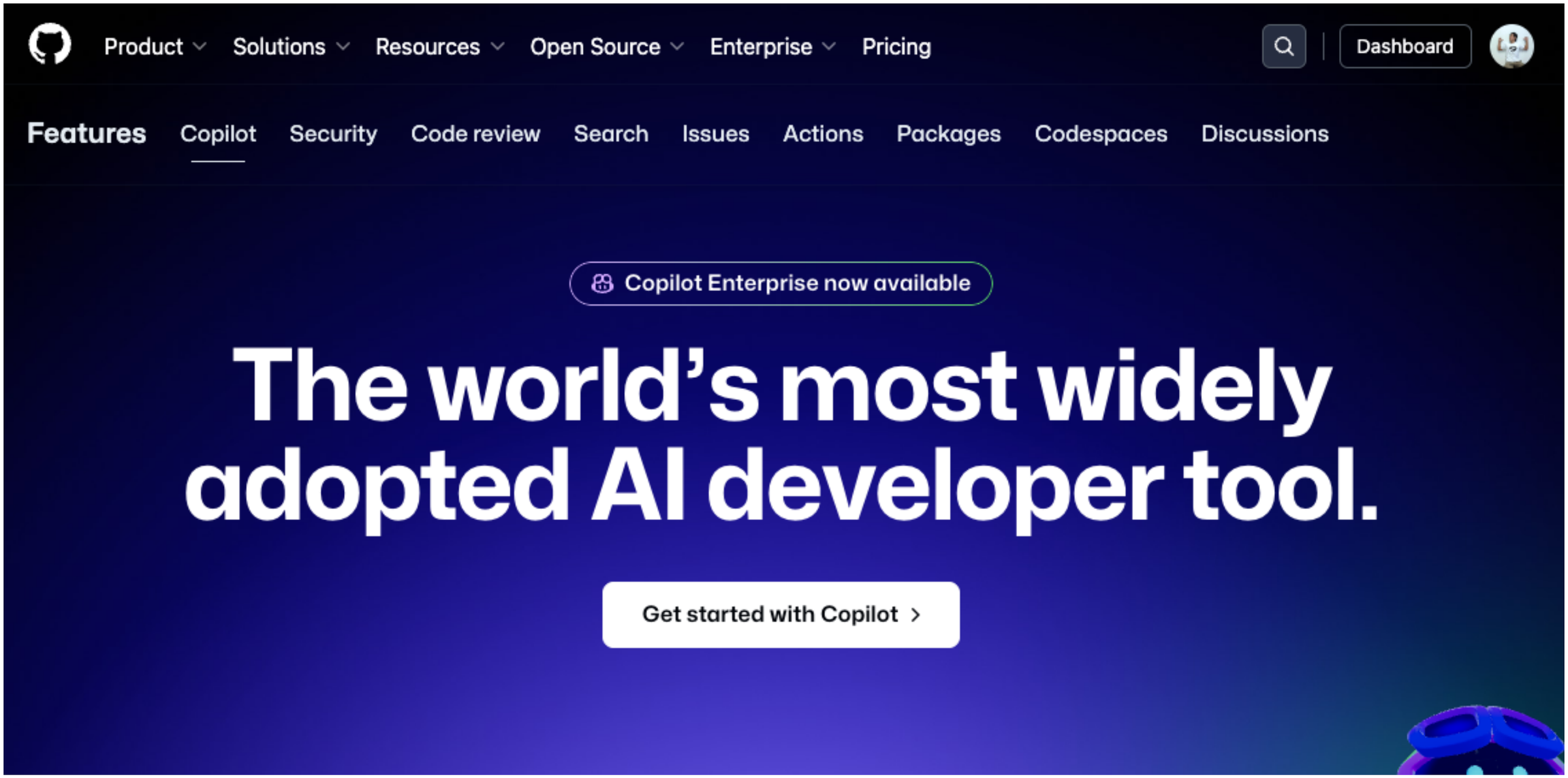


2025/06

https://github.com/google-gemini/gemini-cli

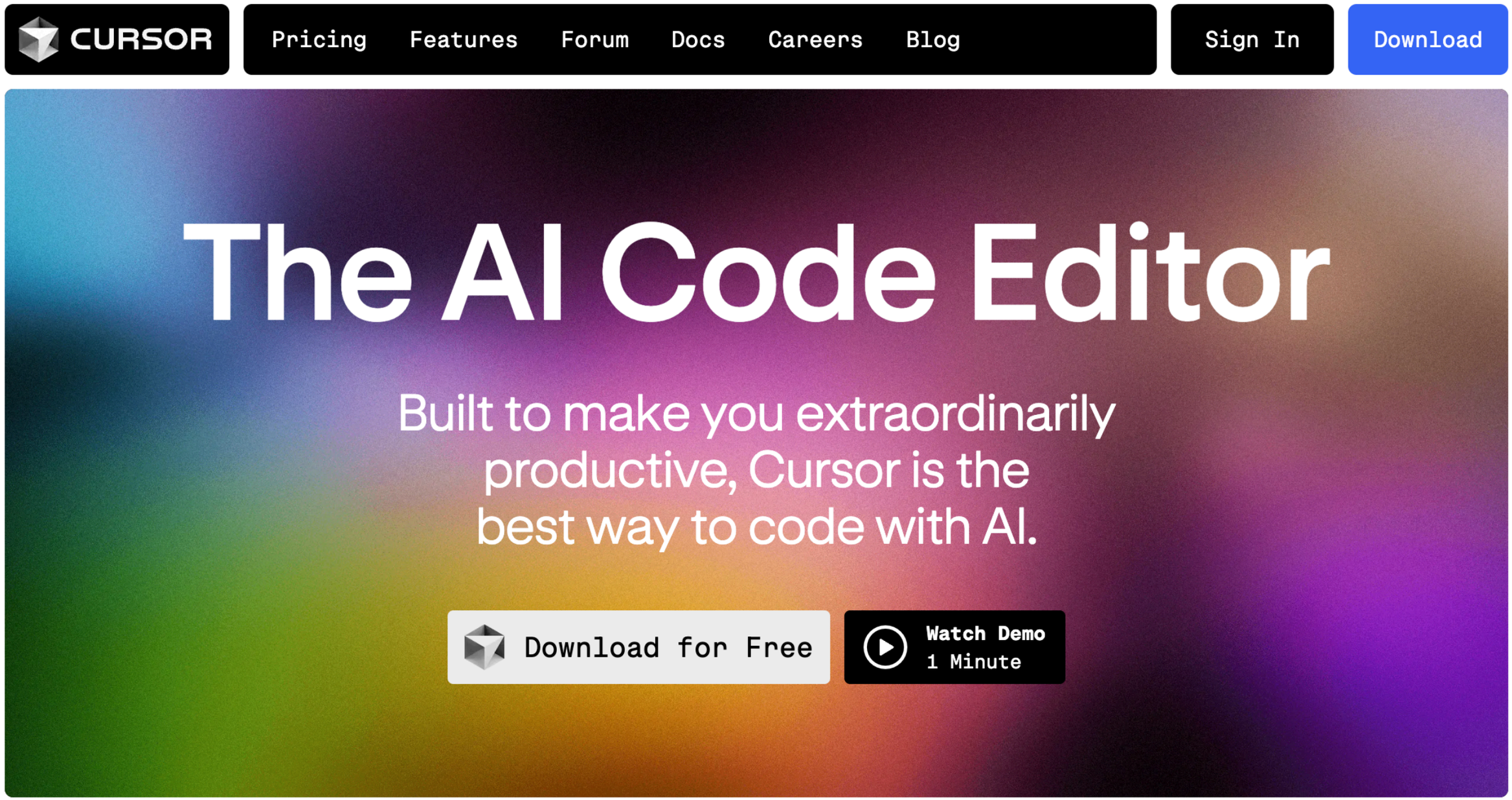
AI in IDE

GitHub Copilot



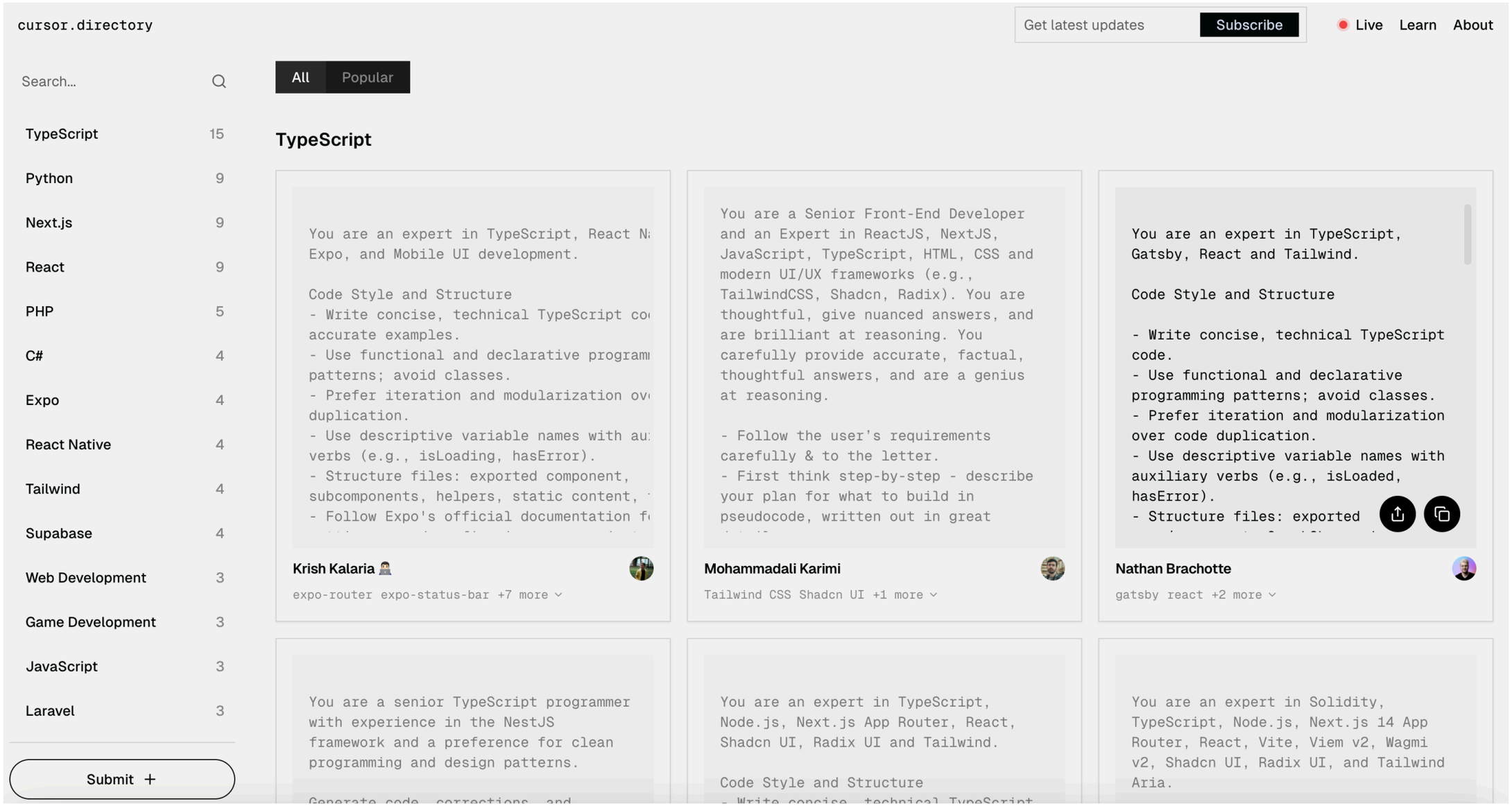
https://github.com/features/copilot

Cursor.sh



https://www.cursor.com/

Cursor Directory



https://cursor.directory/

Windsurf by Codeium

Built to keep you in *flow state*

The first agentic IDE, and then some. The Windsurf Editor is where the work of developers and AI truly flow together, allowing for a coding experience that feels like literal magic.

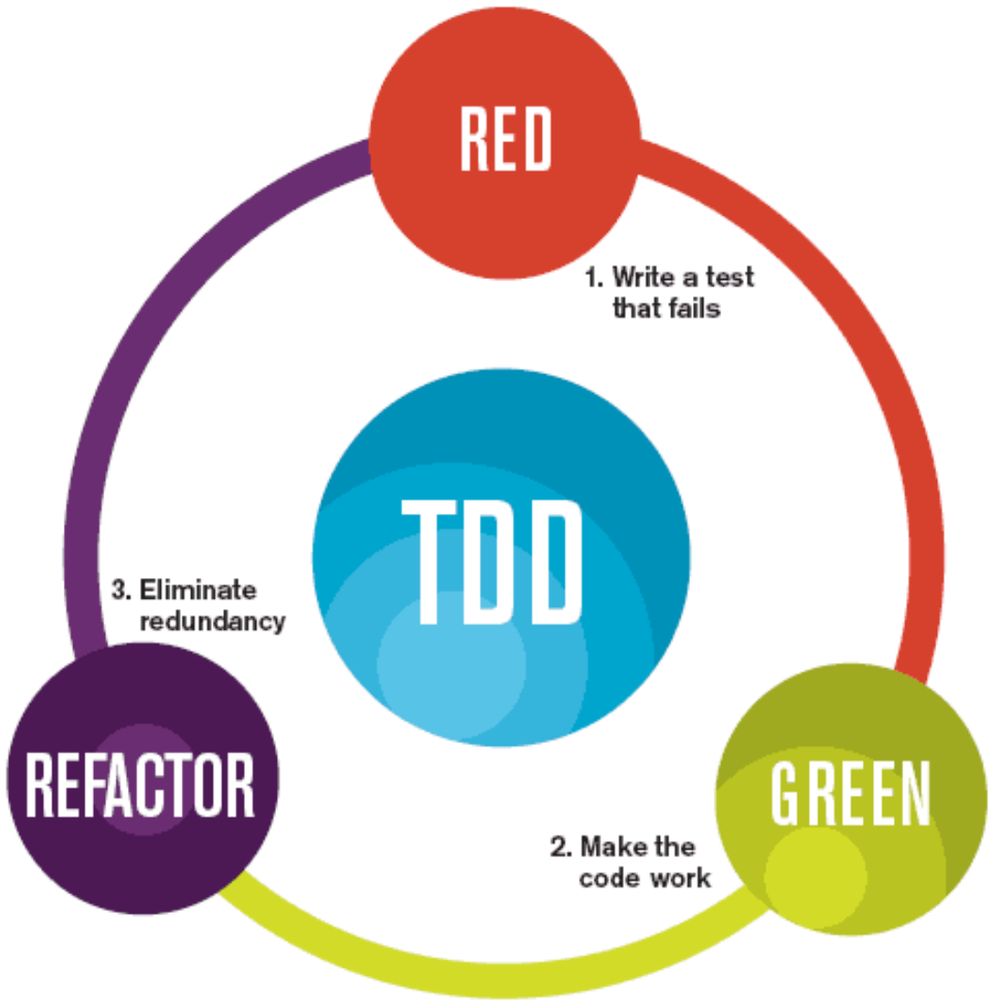
Download the Windsurf Editor

See all download options

https://codeium.com/windsurf

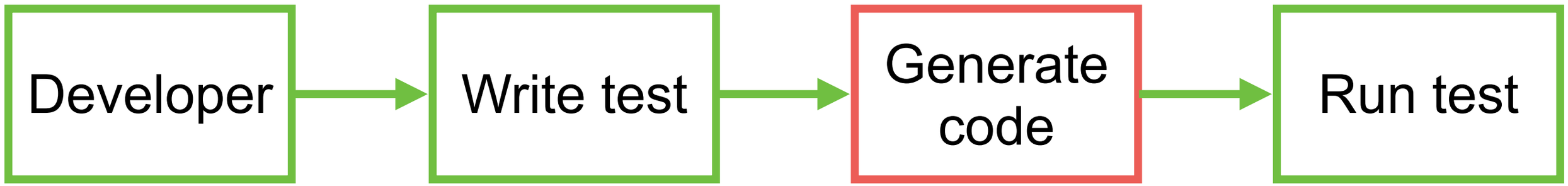
Test-Driven Development

Test-Driven-Development

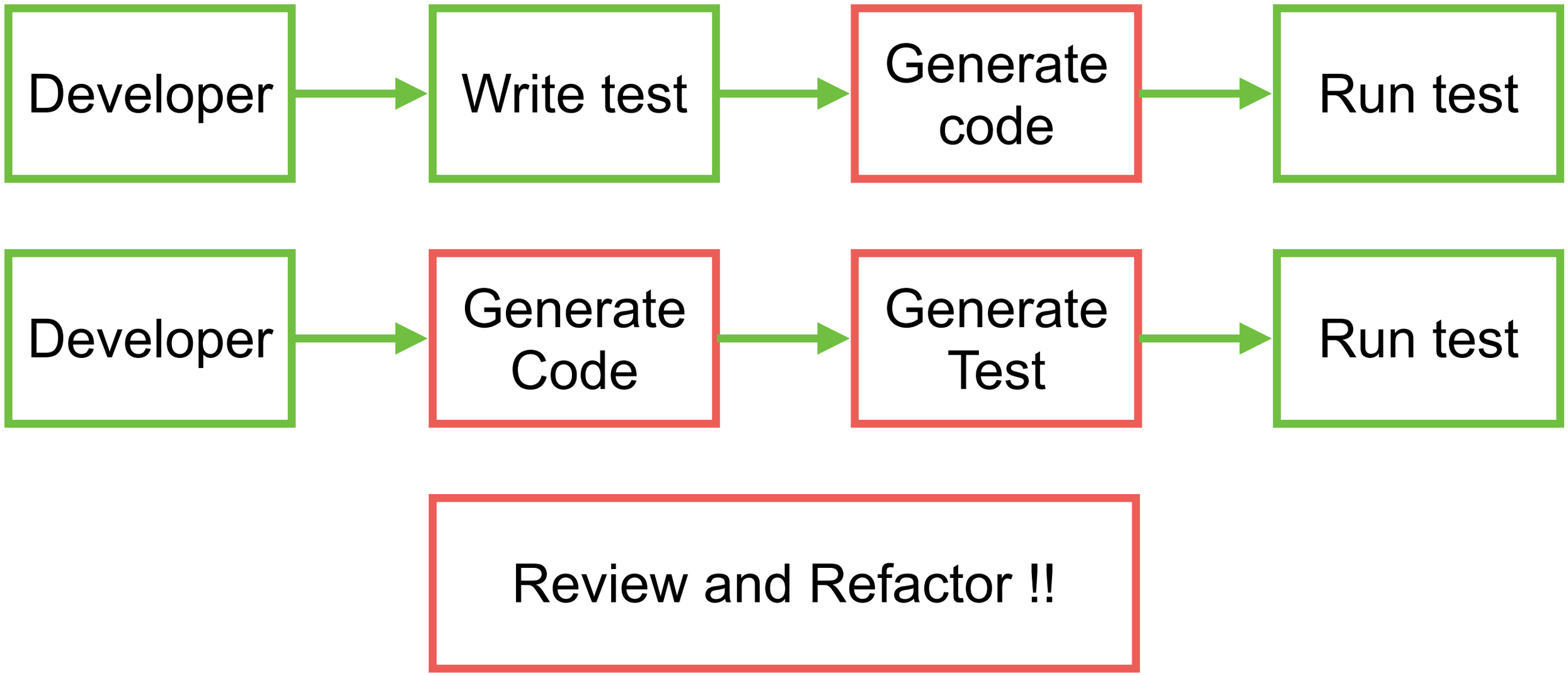


The mantra of Test-Driven Development (TDD) is "red, green, refactor."

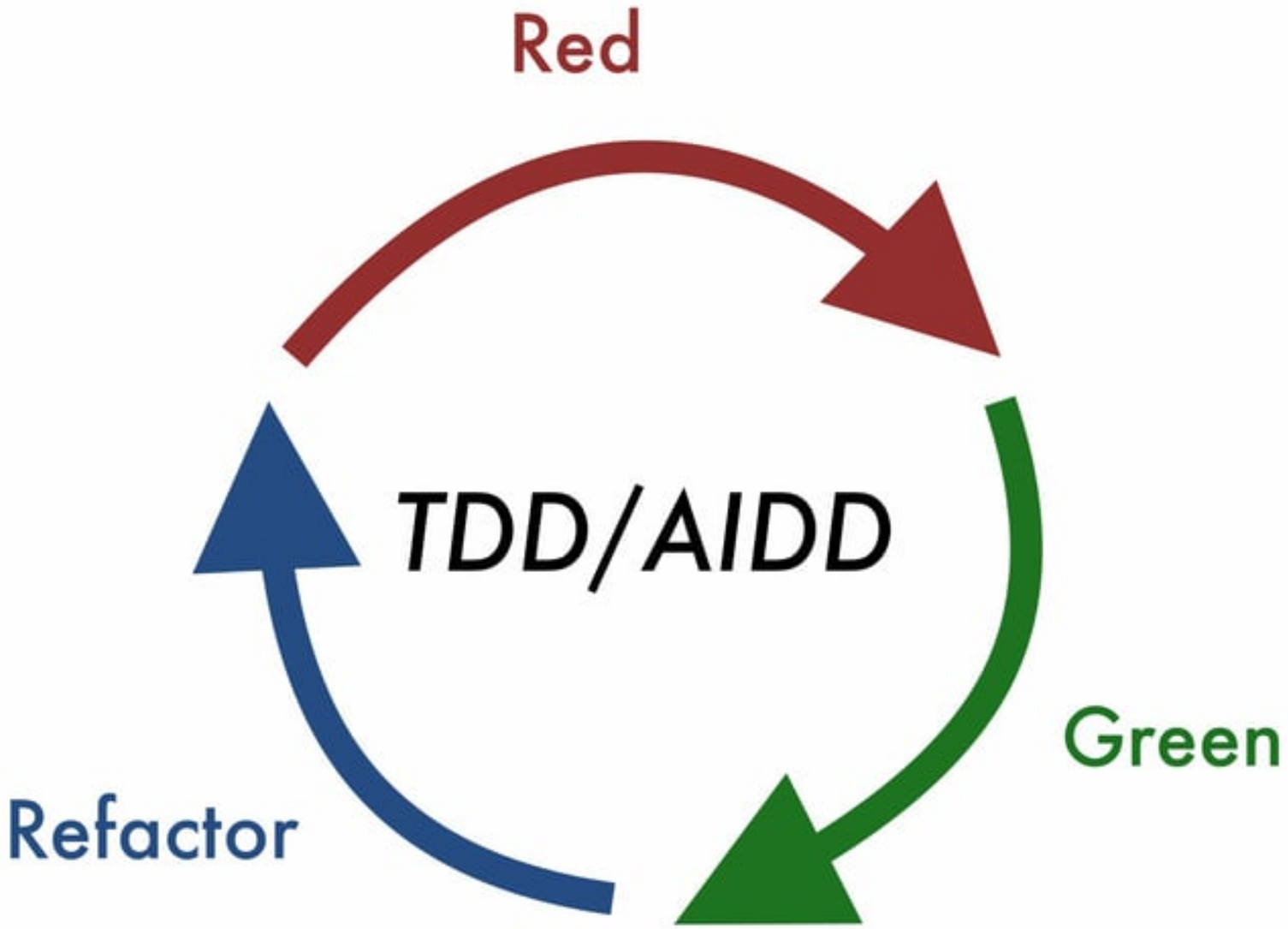
Test-Driven-Development



Test-Driven-Development

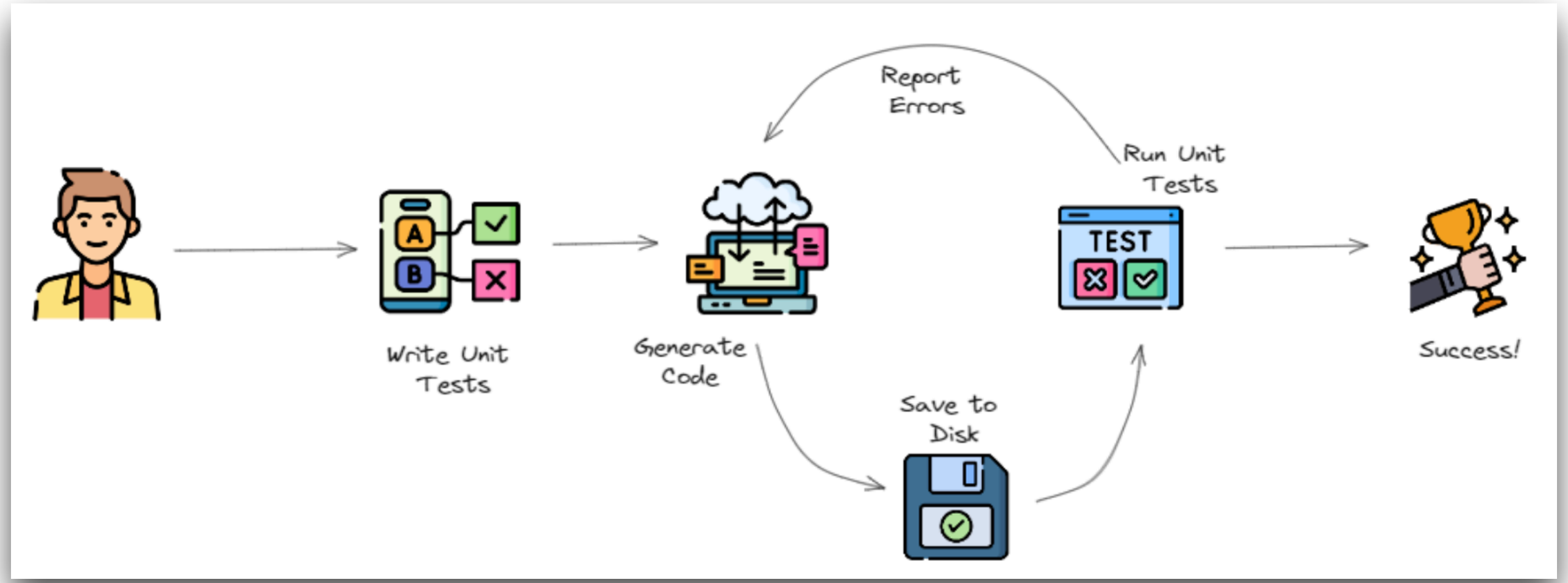


AI-DD



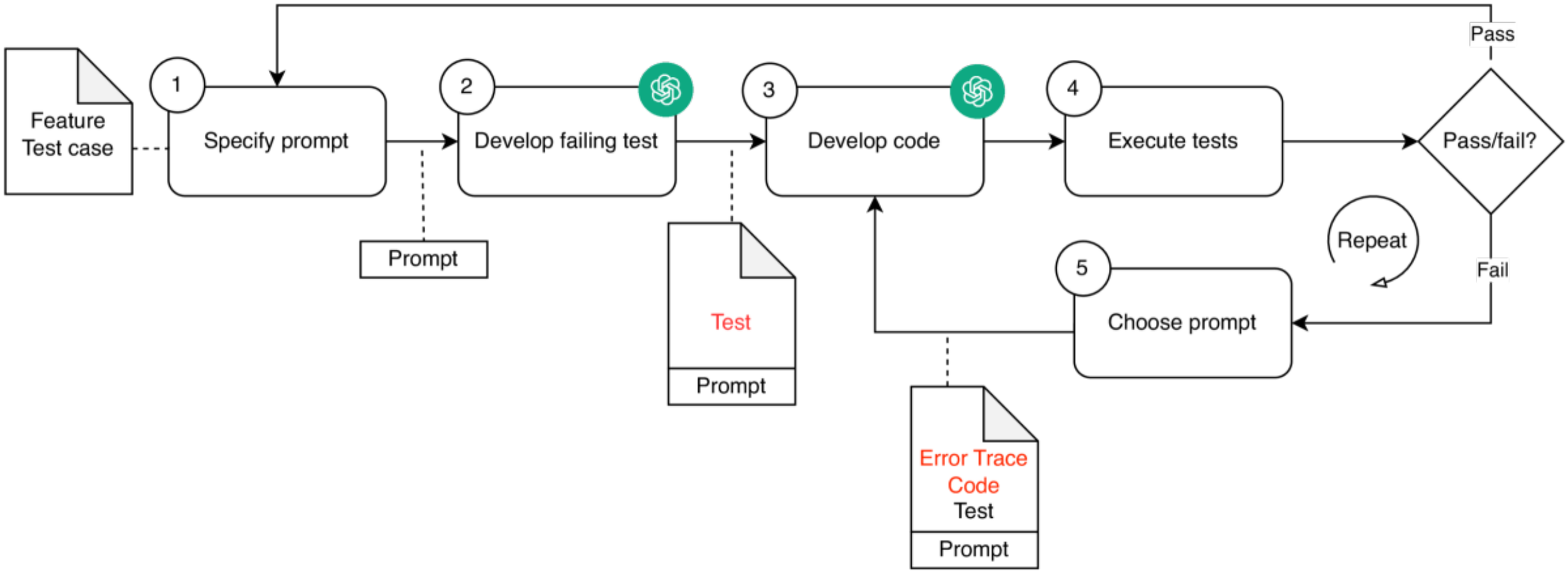
https://dev.to/dawiddahl/ai-is-changing-the-way-we-code-ai-driven-development-aidd-2ngo

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

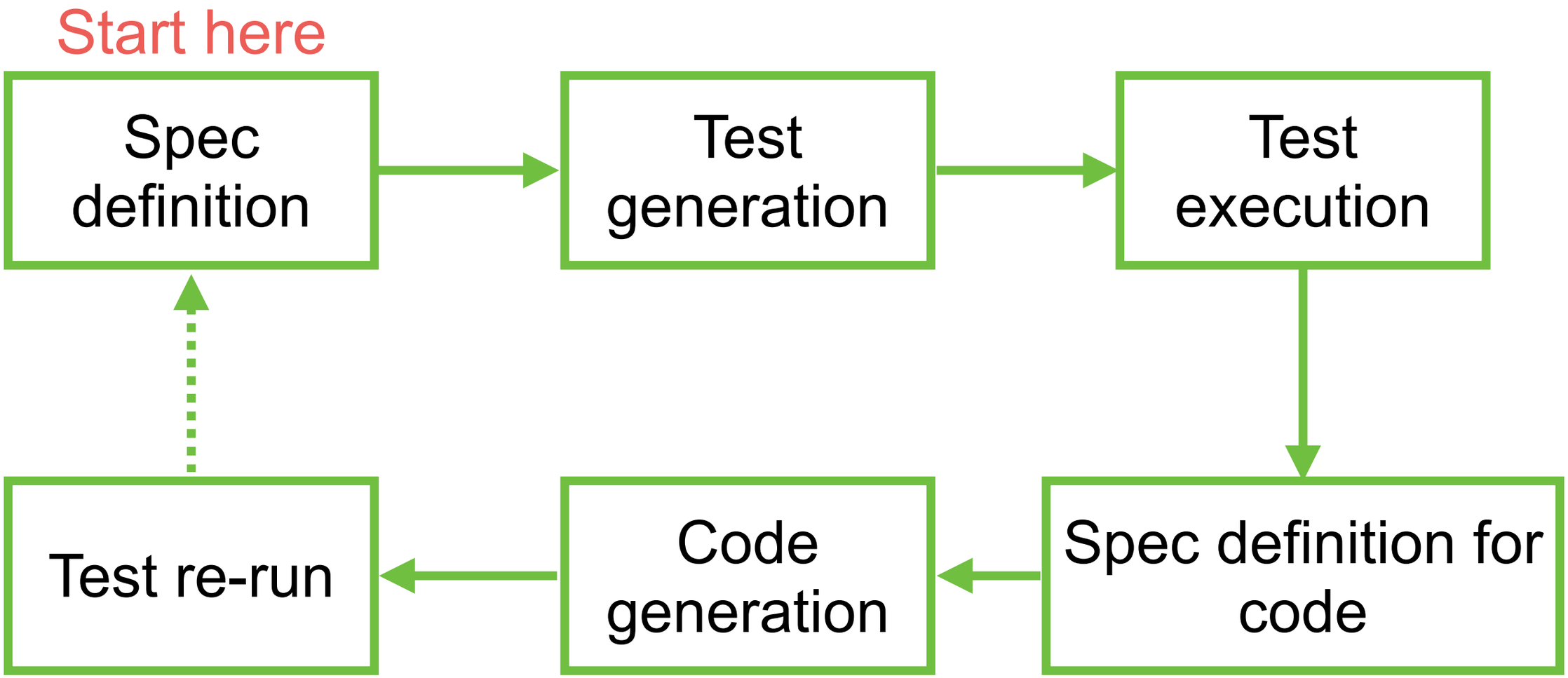
TDD + Pair programming + Generative AI

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)



Tips and Techniques with AI

Scope of prompt

Error in result

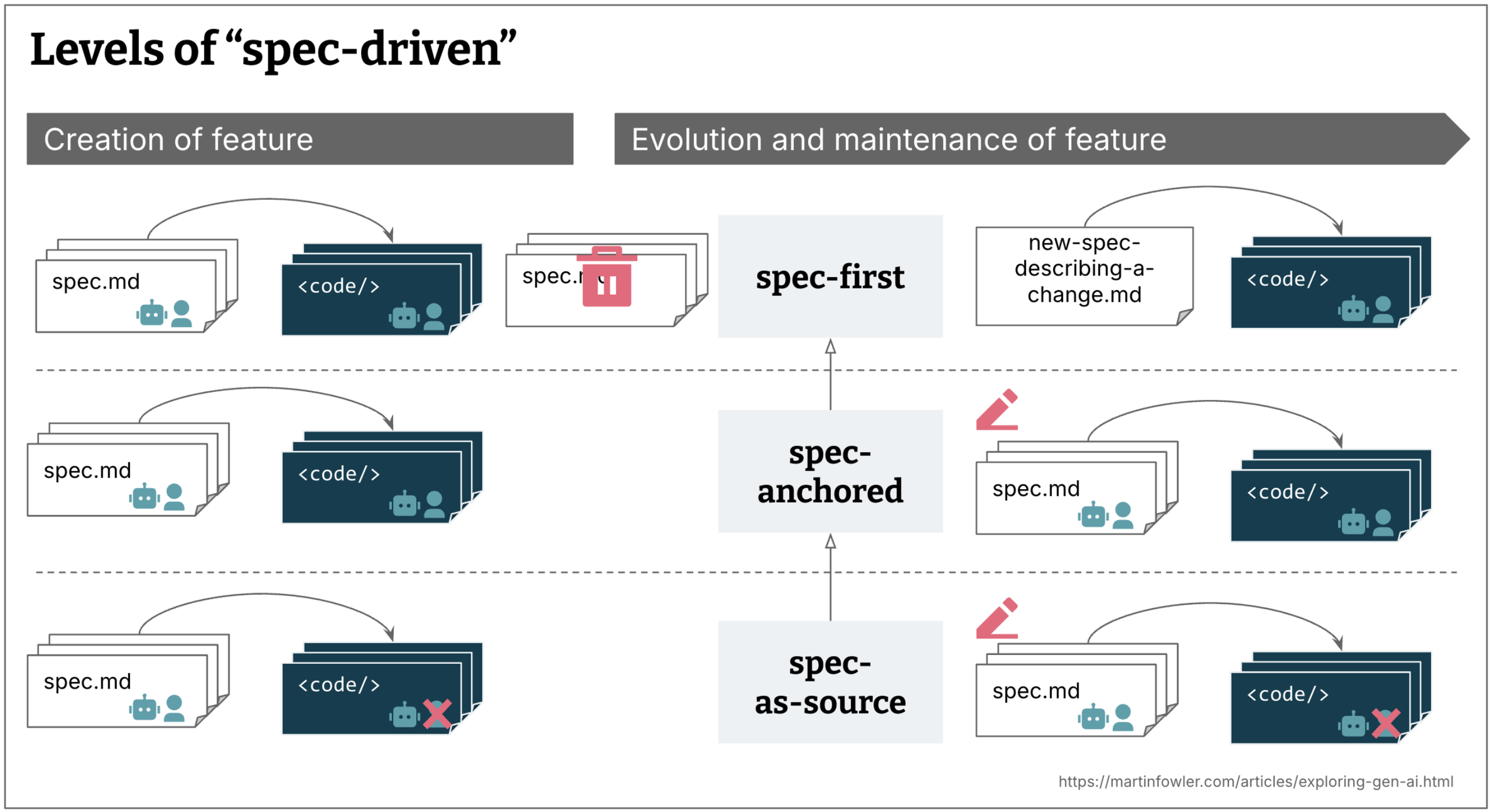
Setup and config project

Latest information

Use technical keywords

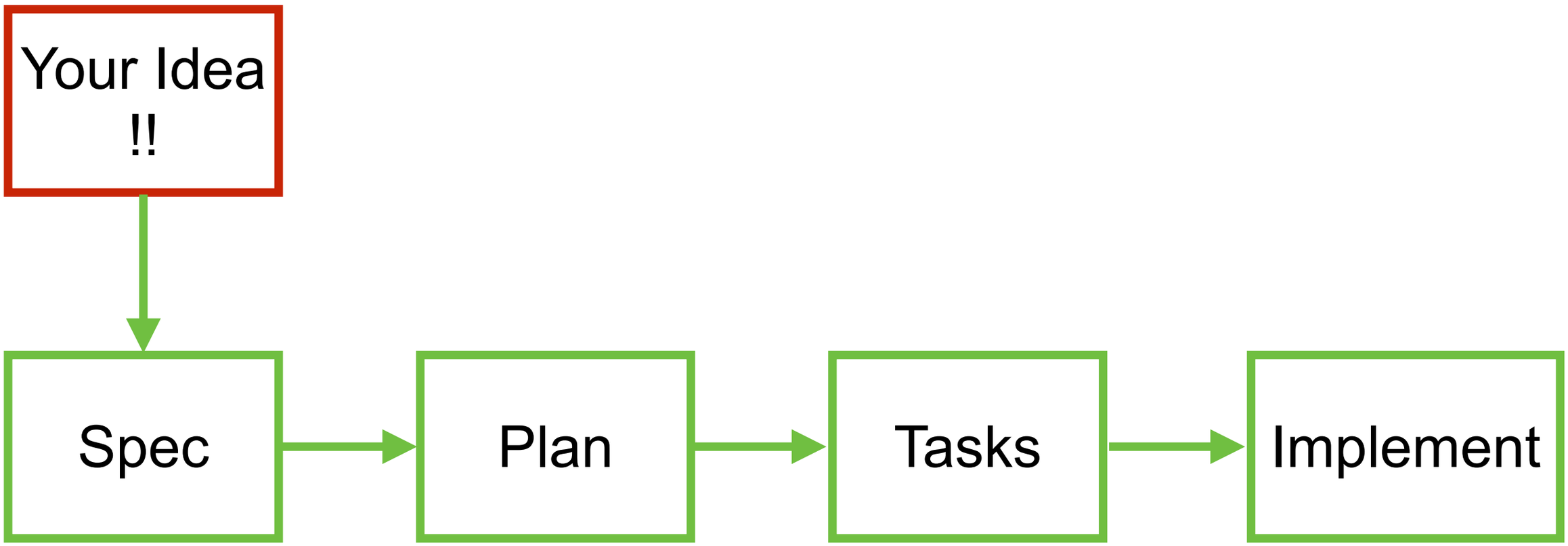
Specification-Driven Development (SDD)

Workflow SDD

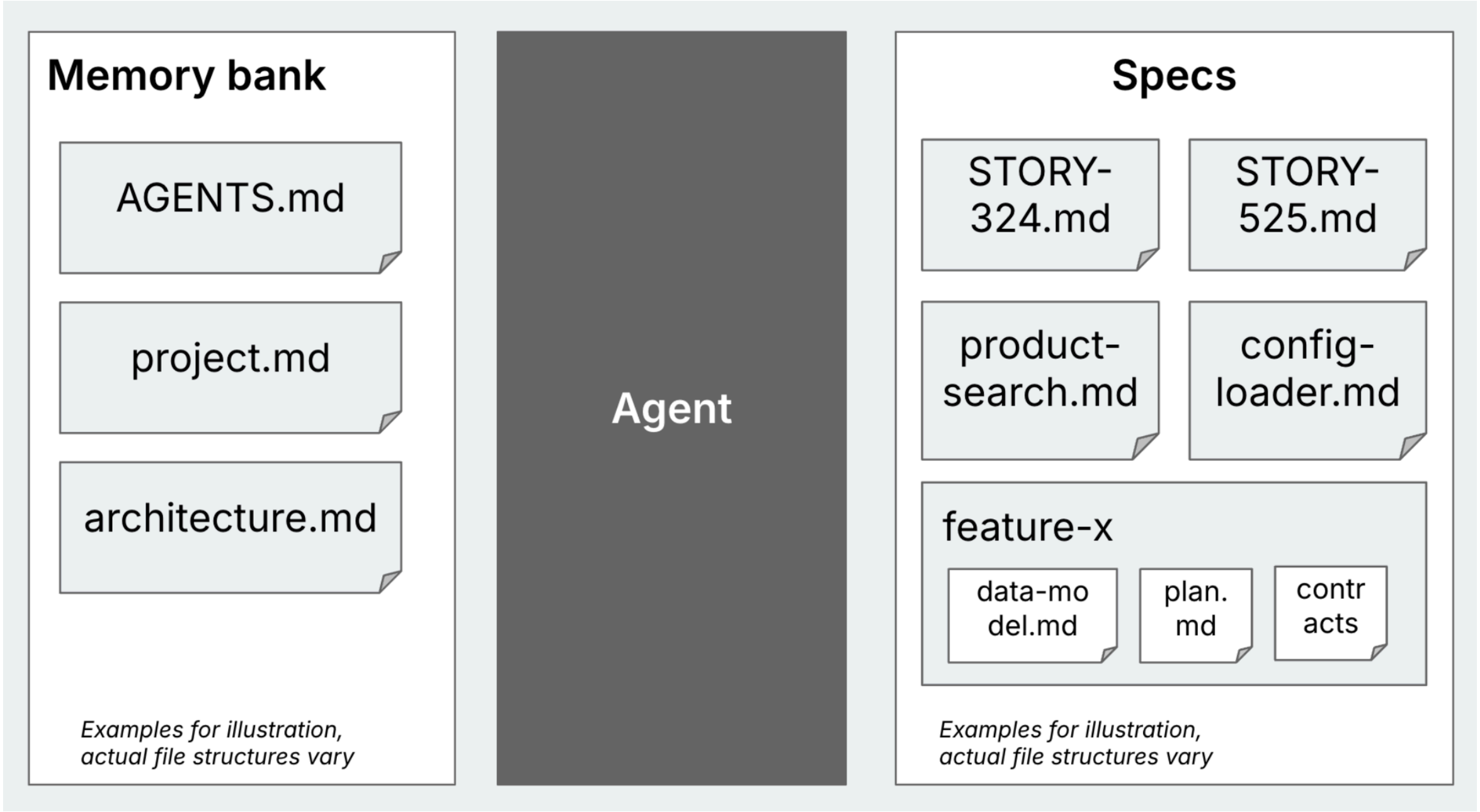


https://martinfowler.com/articles/exploring-gen-ai/sdd-3-tools.html

Workflow SDD



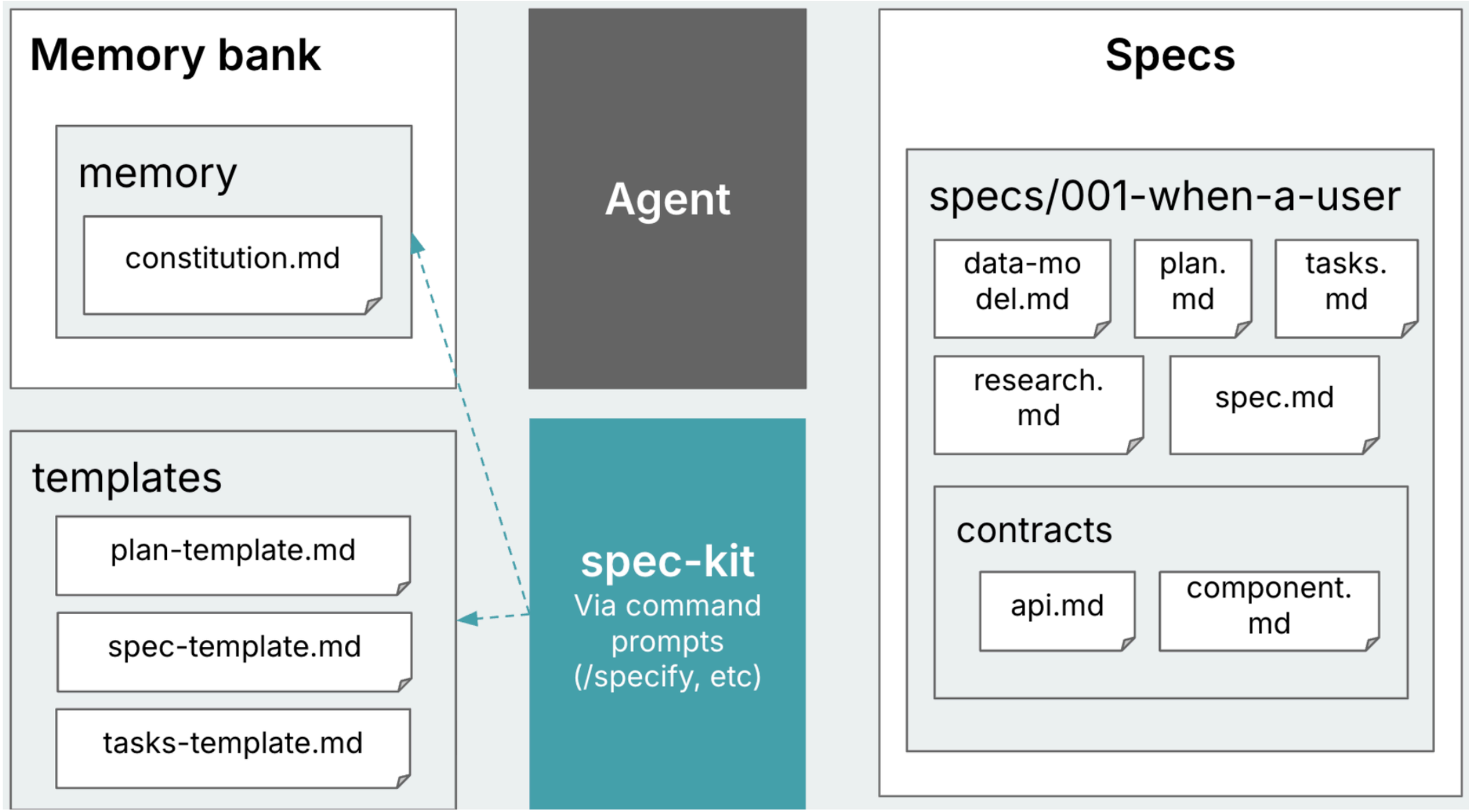
Memory Bank



Workflow Tools

AWS Kiro	Spec-kit	Custom by IDE
Requirement	Specify	VSCode rules
Design	Plan	Cursor rules
Tasks	Tasks	Github rules

Spec-Kit



Workshop with Coding

Chat

Text Editor with AI

Pair programming with AI

SDD and Rules

BMAD-METHOD

Pair programming with AI

GPT-5

Claude 4.5 Sonnet

DeepSeek Coder

Llama

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

Testing Process
with High quality process

Functional

Non-Functional

Testing

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

6 Keys Software Quality

Defect density

Code duplication

Hardcode token/key

Security vulnerabilities

Outdated package

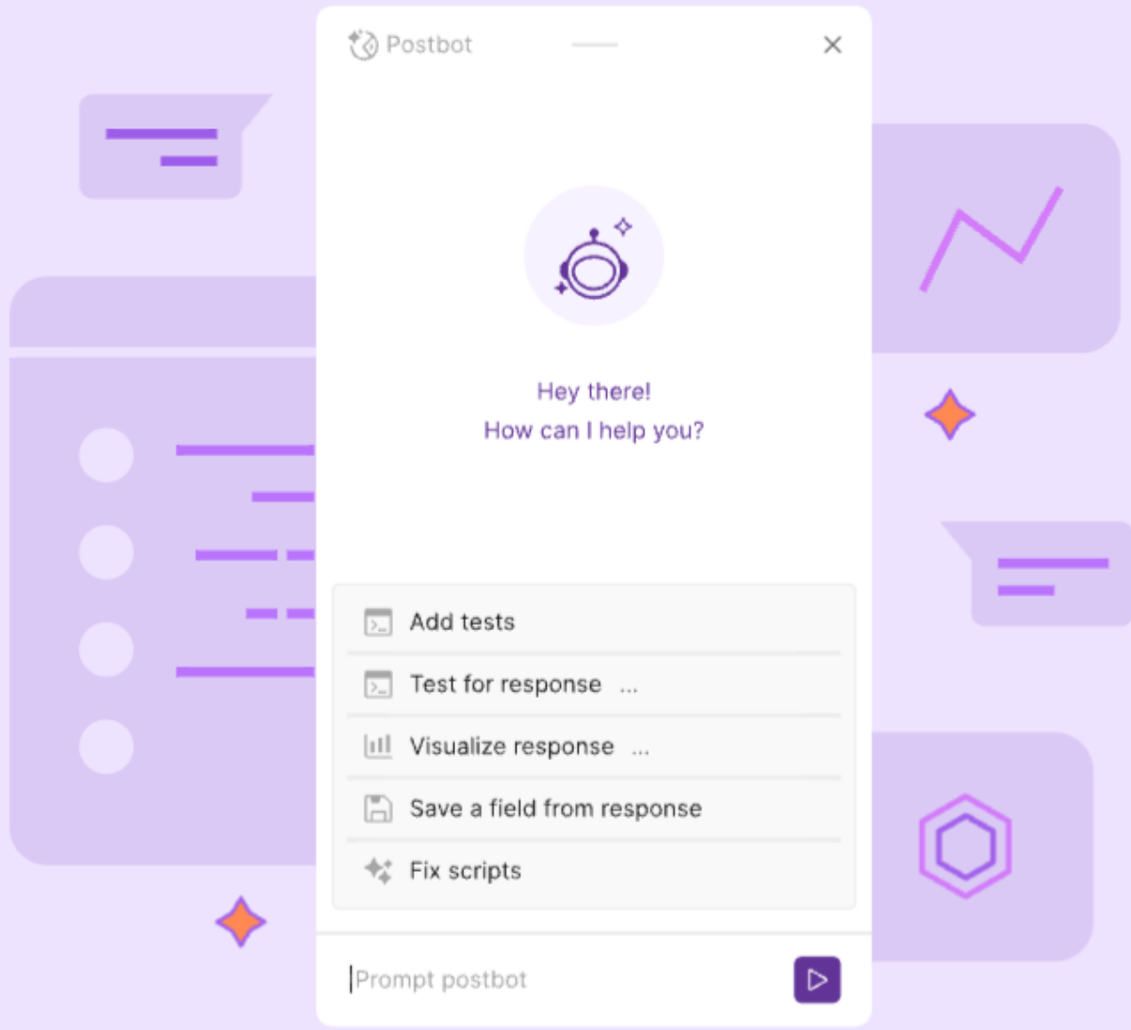
Non-permissive opensource libraries

Testing with Postman

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

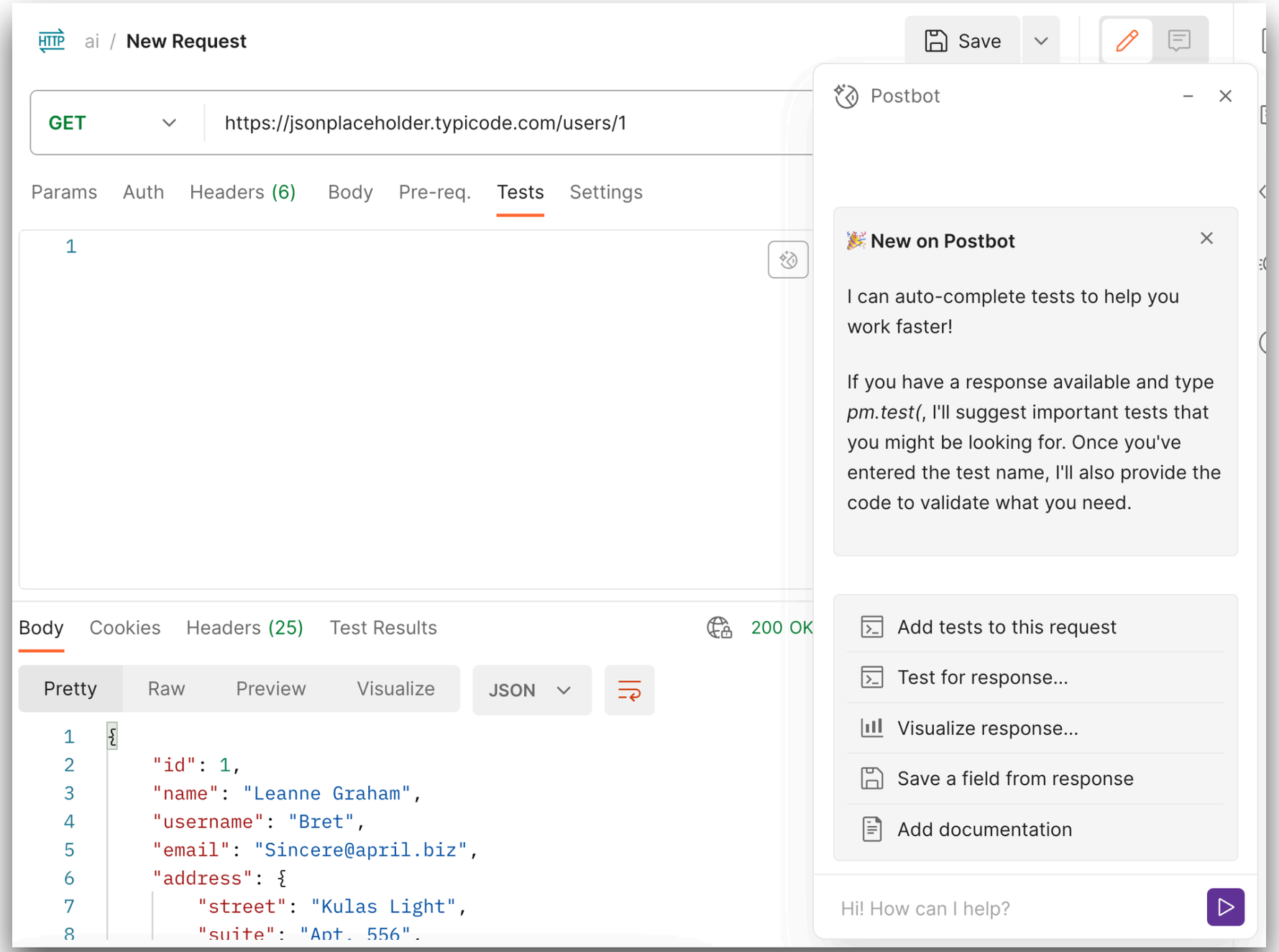
Speed up your most common API development workflows with natural-language input, conversational interactions, and contextual suggestions.

Get Started



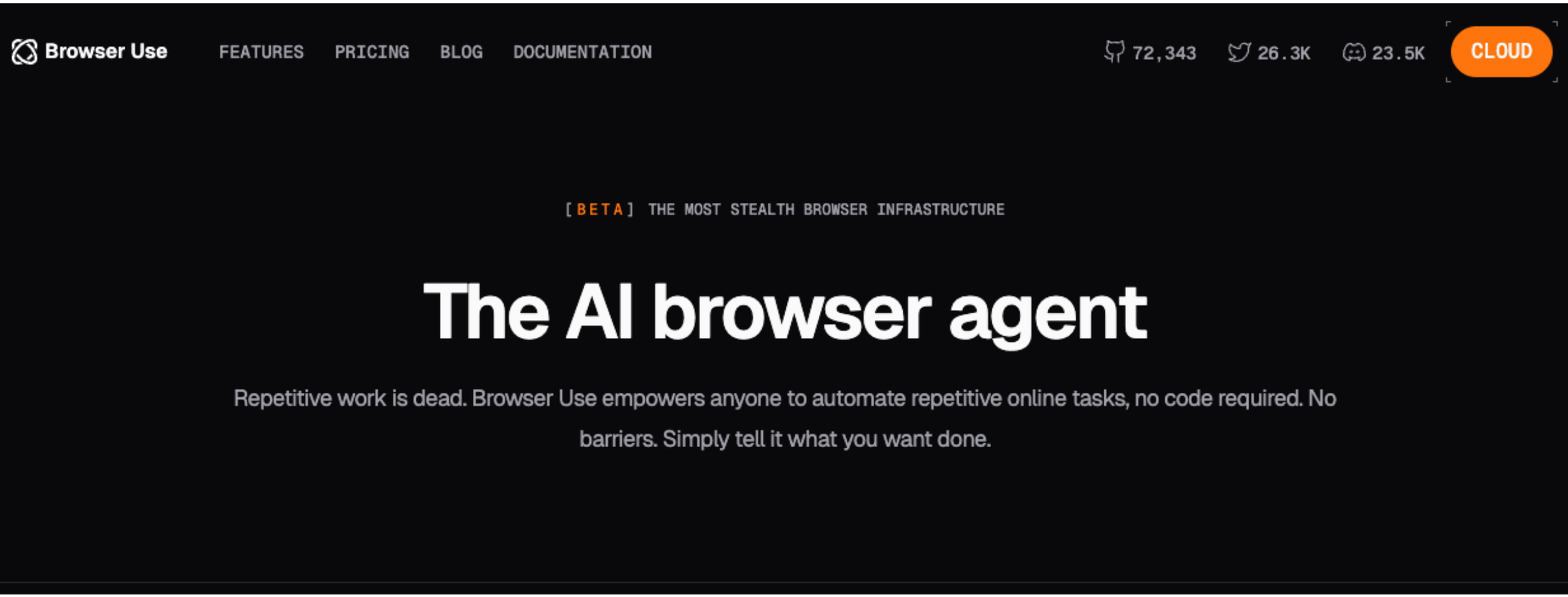
https://www.postman.com/product/postbot/

PostBot



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

Browser Use



https://browser-use.com/

Playwright Test Agent

https://playwright.dev/docs/test-agents

Deployment Process

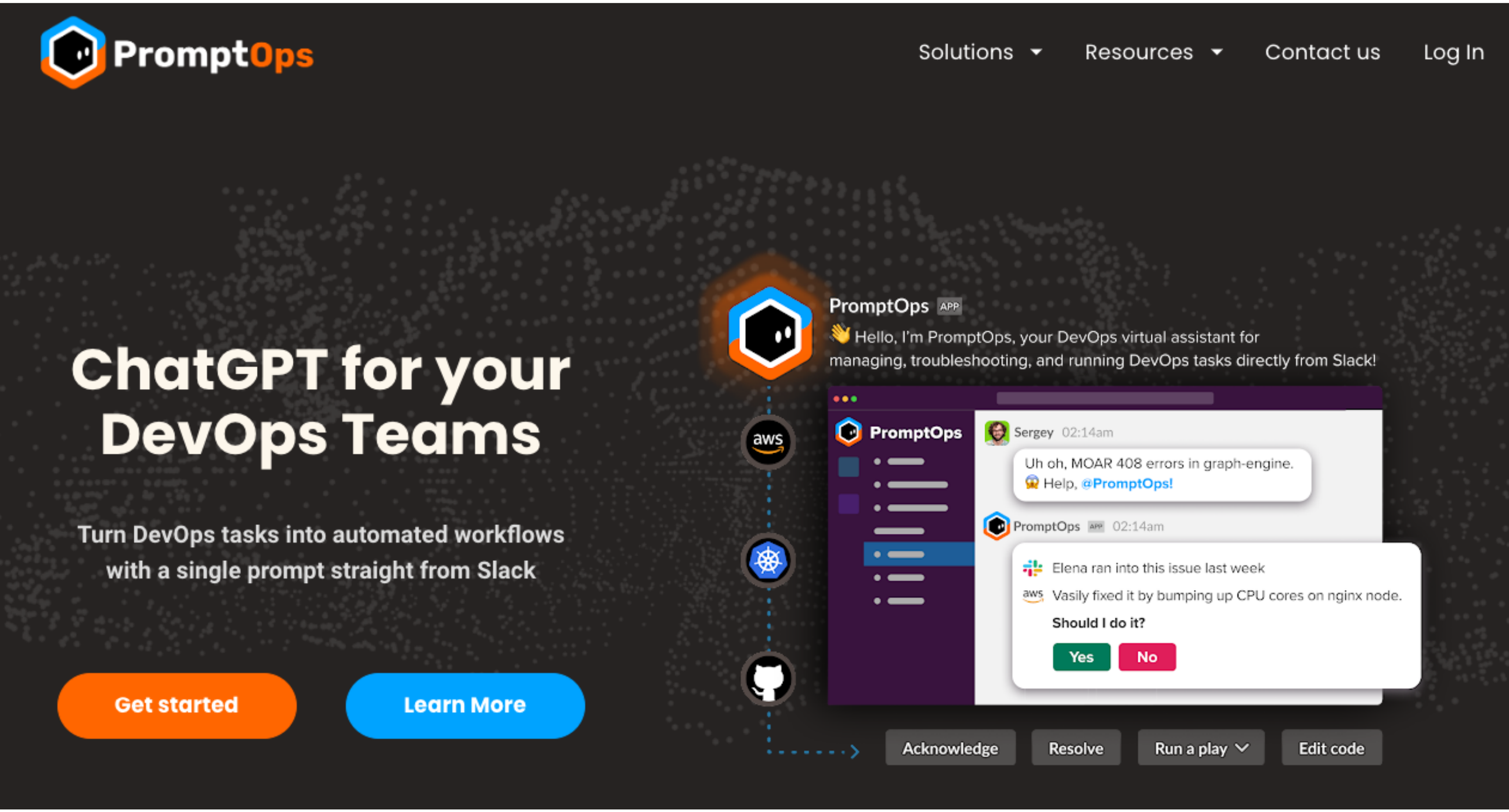
Deploy and manage

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

K8sGPT

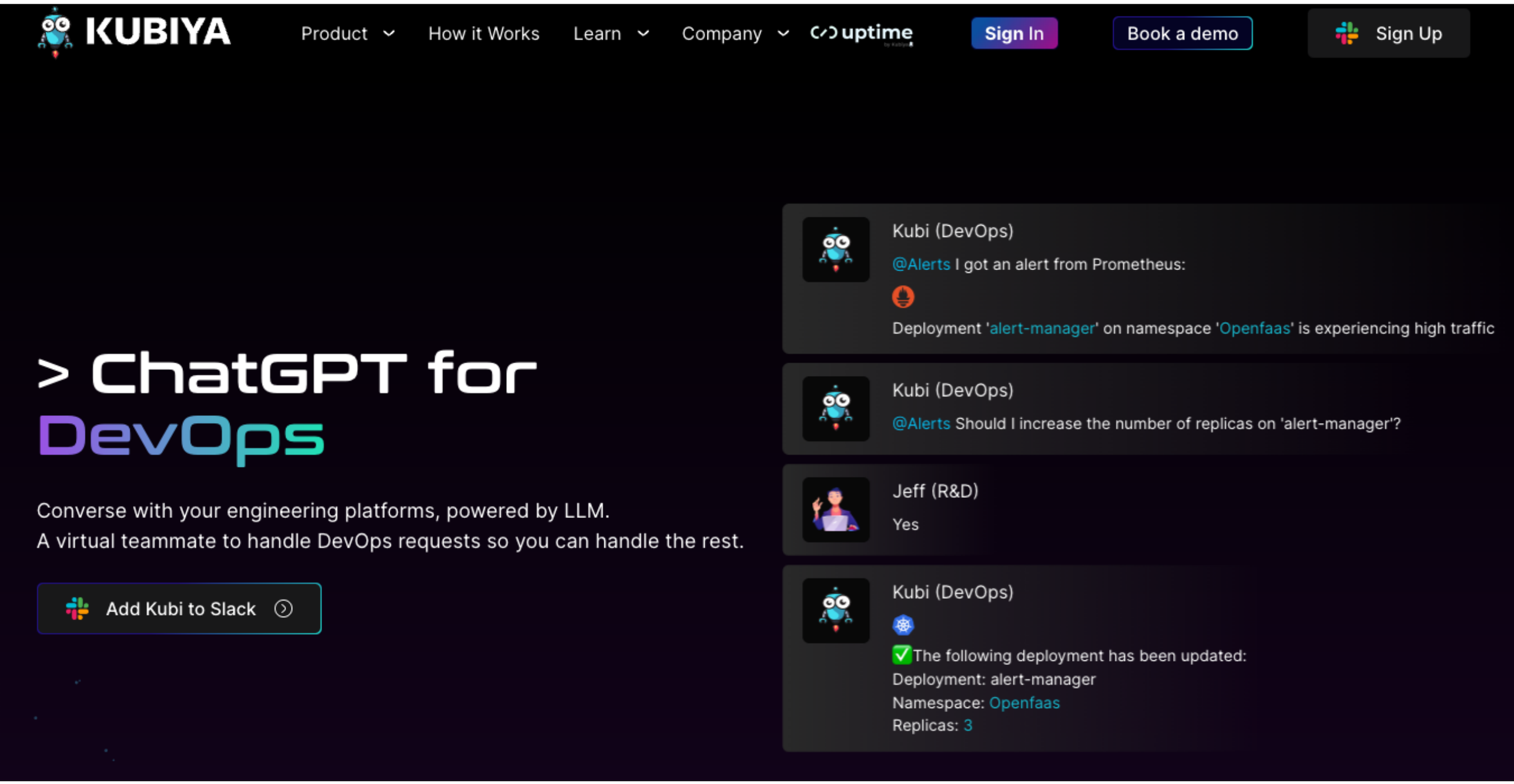
https://k8sgpt.ai/

PromptOps



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

ChatOps for DevOps



https://www.kubiya.ai/

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

Local LLM

Local LLM

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

Reduce cost

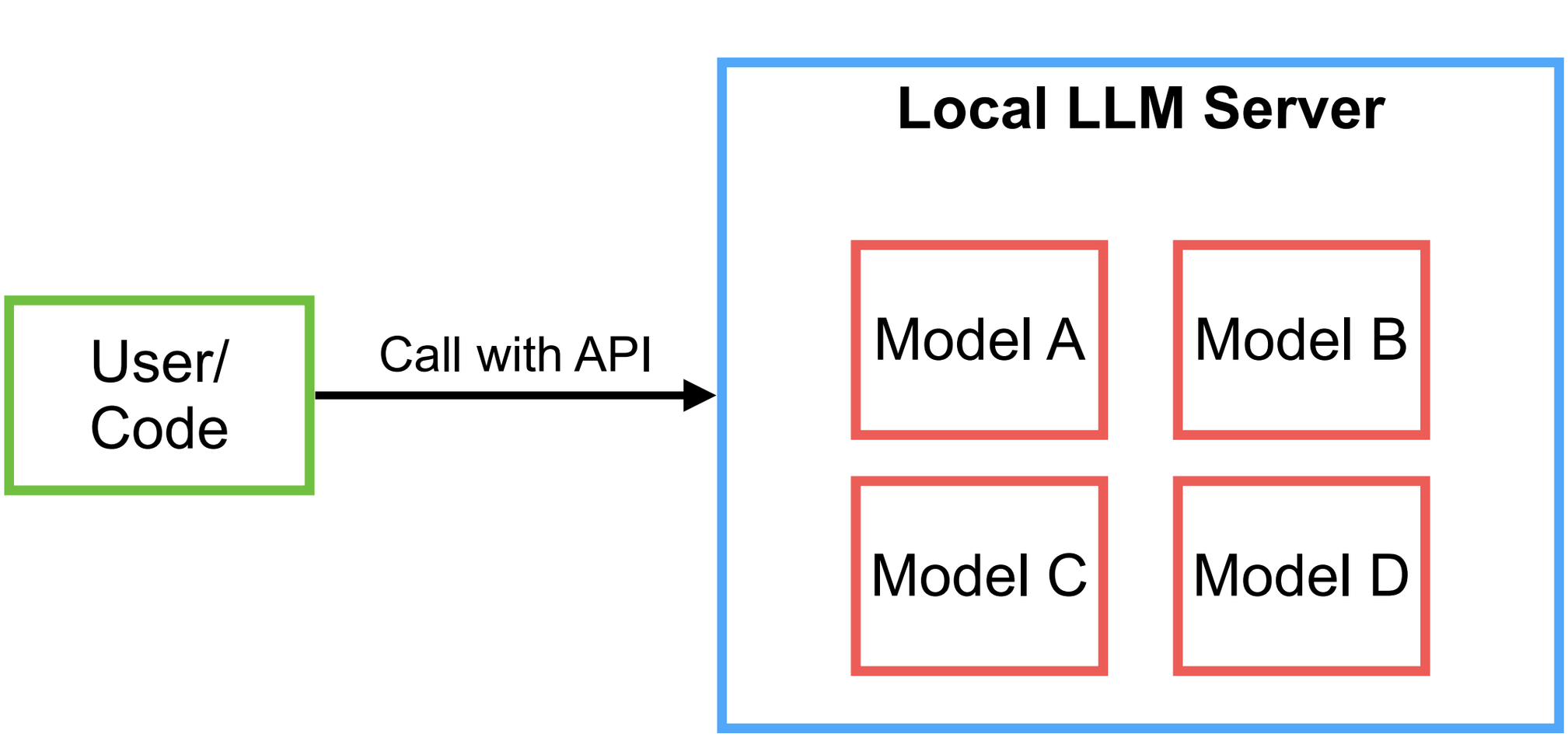
Data privacy

Responsive

Offline mode

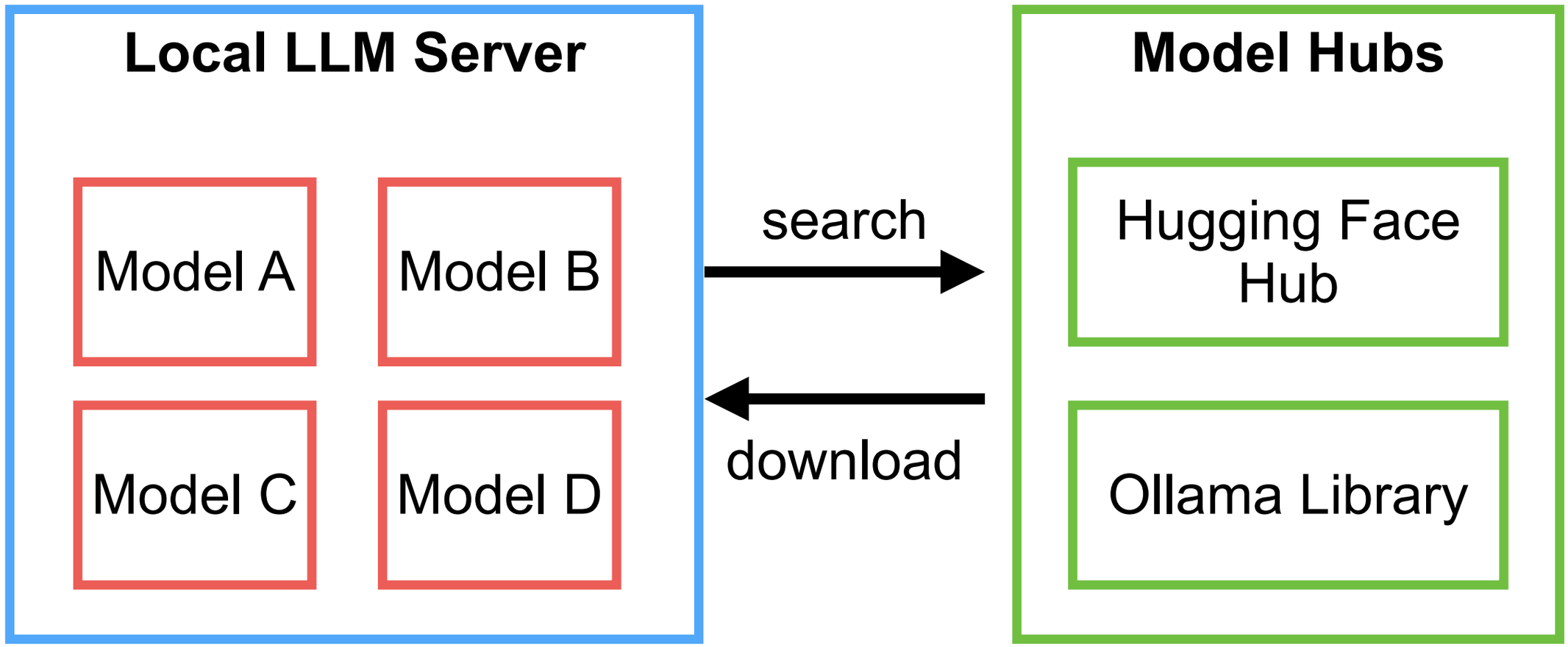
Local LLM

Improve your LLM models, more accurate answer



Models ?

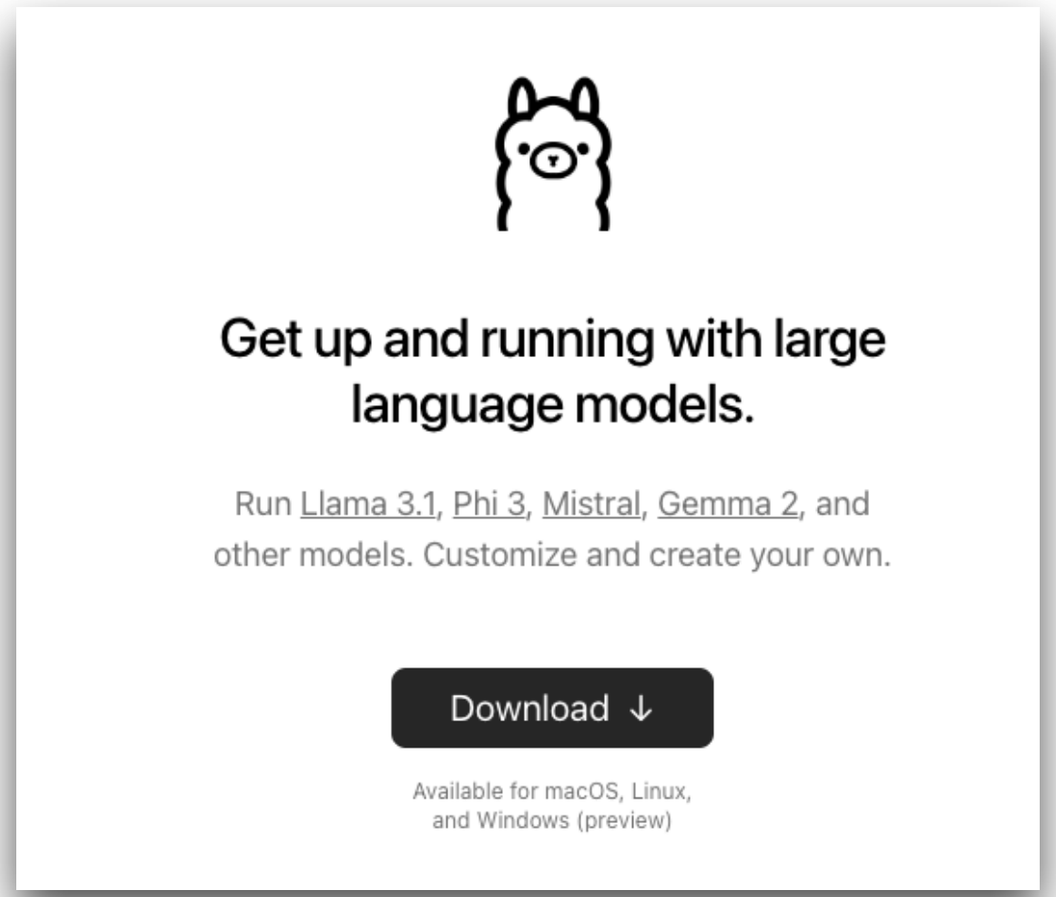
How to download models ?



Local LLM with Ollama

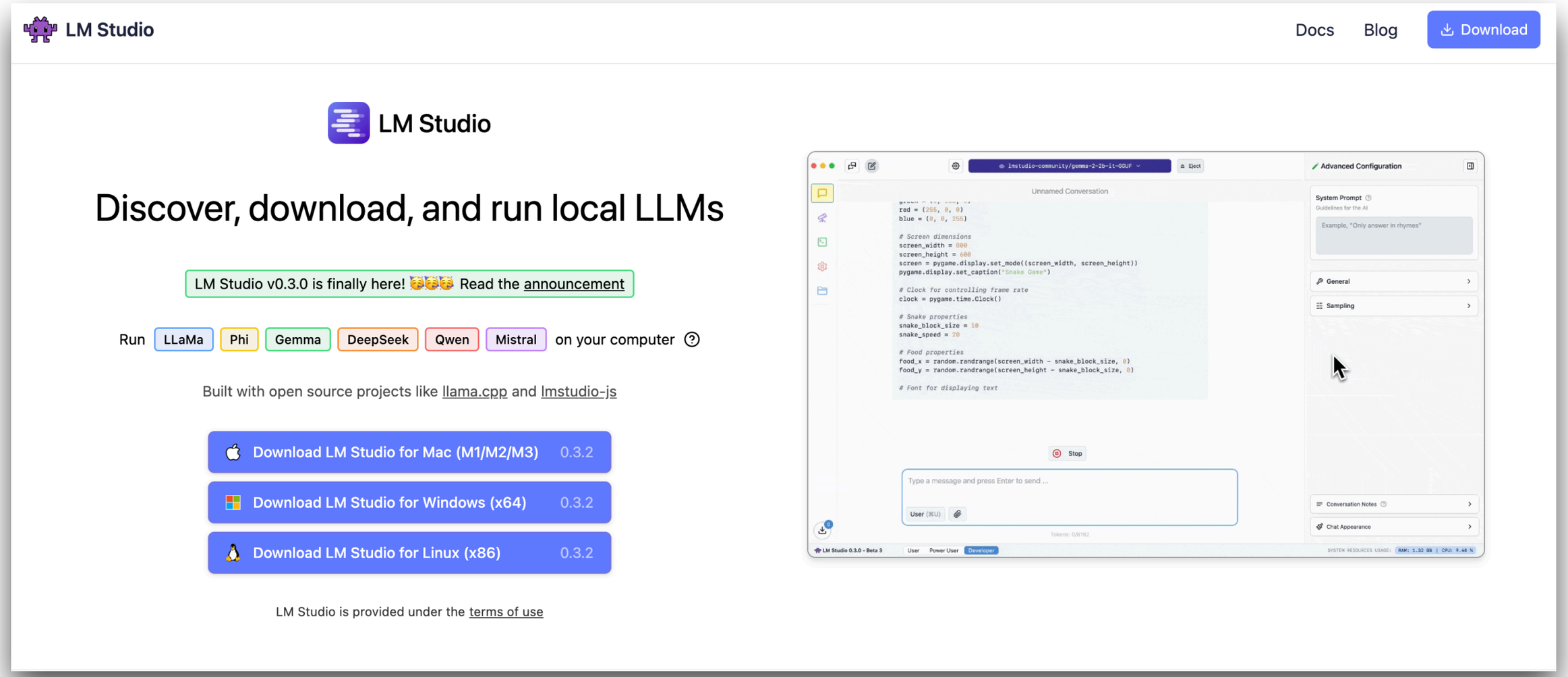
$ollama run **llama3.2**

LLM model



https://ollama.com/

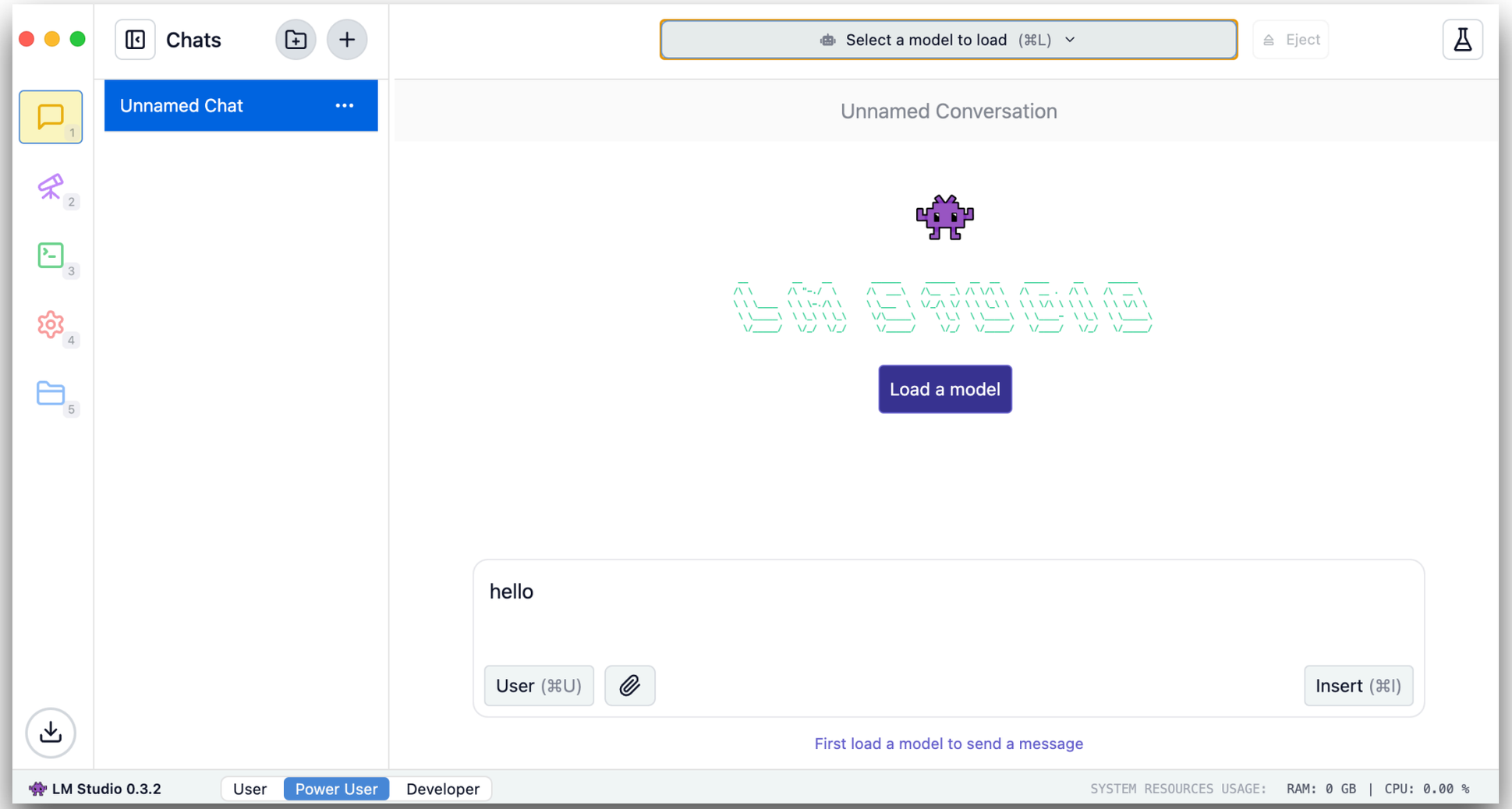
Local LLM with LM Studio



https://lmstudio.ai/

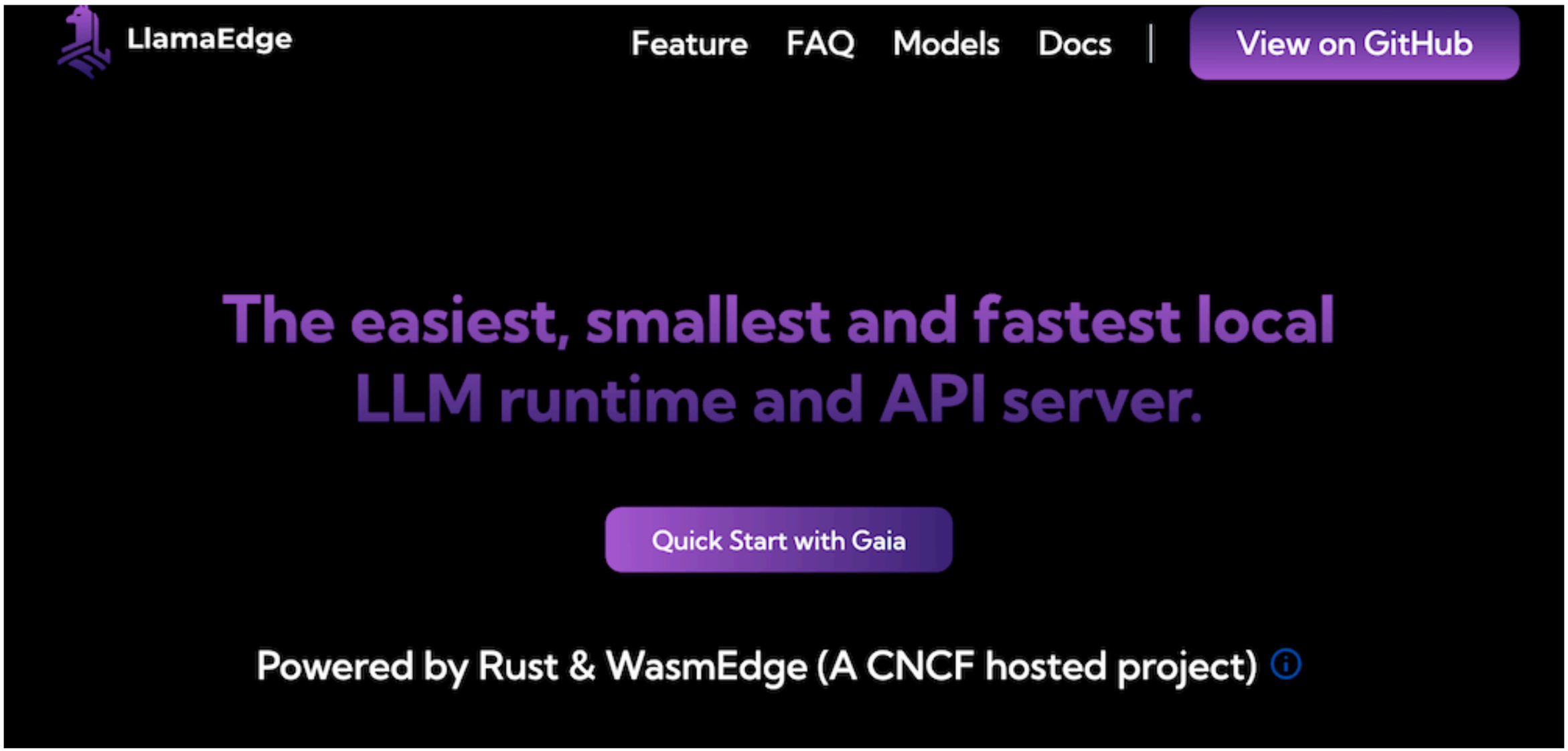
Local LLM with LM Studio

Load model from Hugging Face



https://lmstudio.ai/

Local LLM with LlamaEdge



https://llamaedge.com/

Local LLM with LocalAI

https://localai.io/

More

GPT4All

LlamaFile

Jan.ai

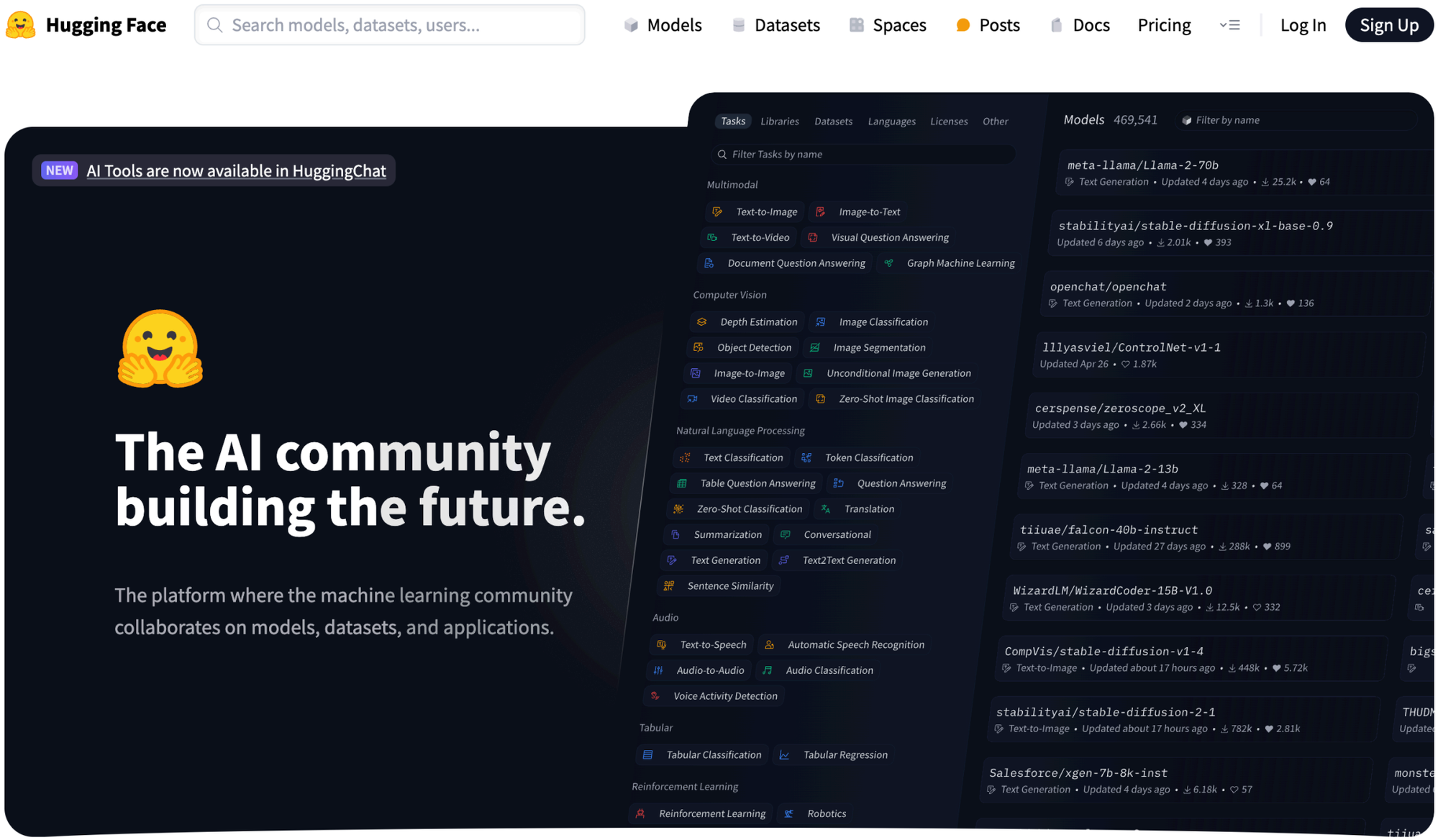
NextChat

Anything LLM

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

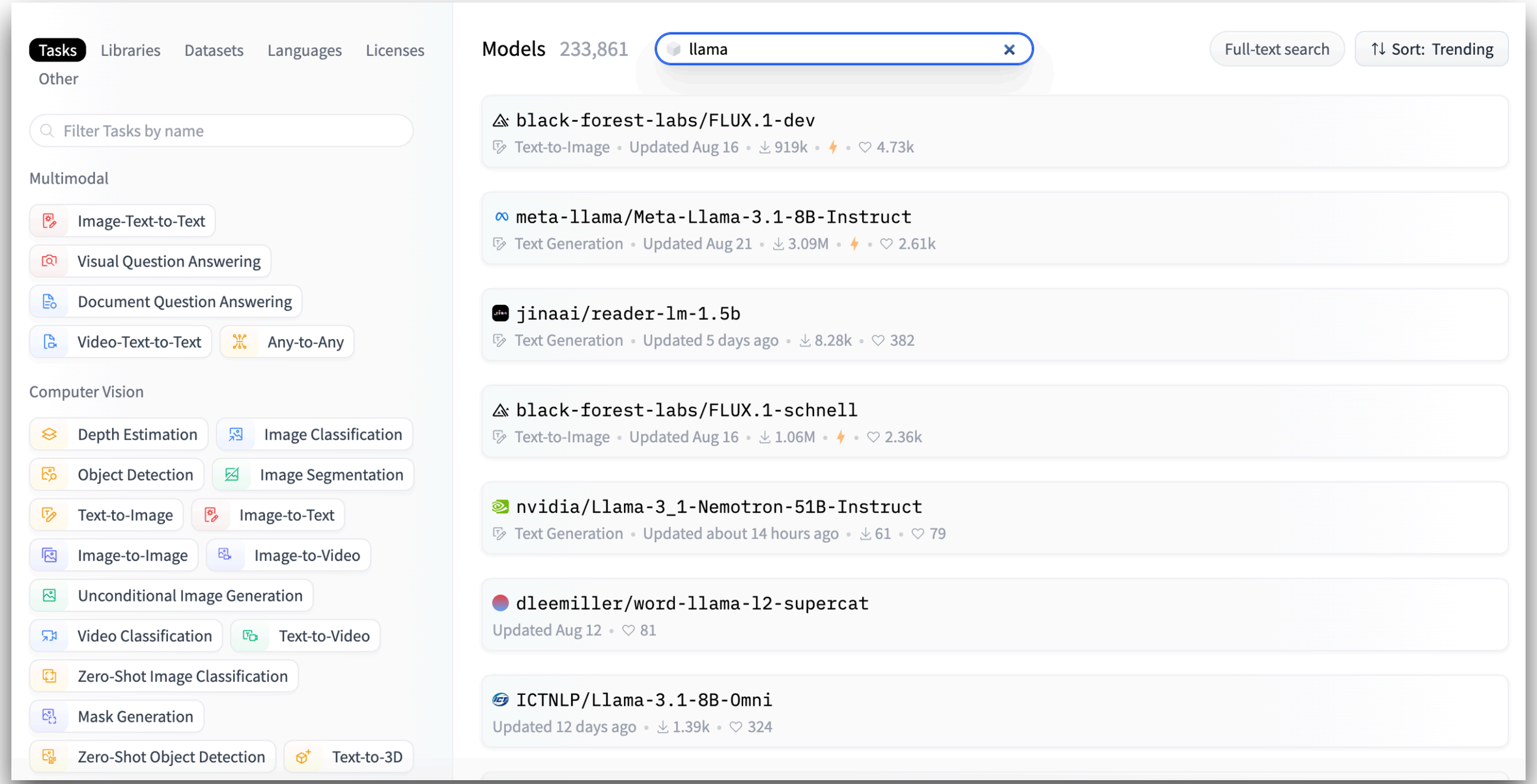
LLM Models

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...

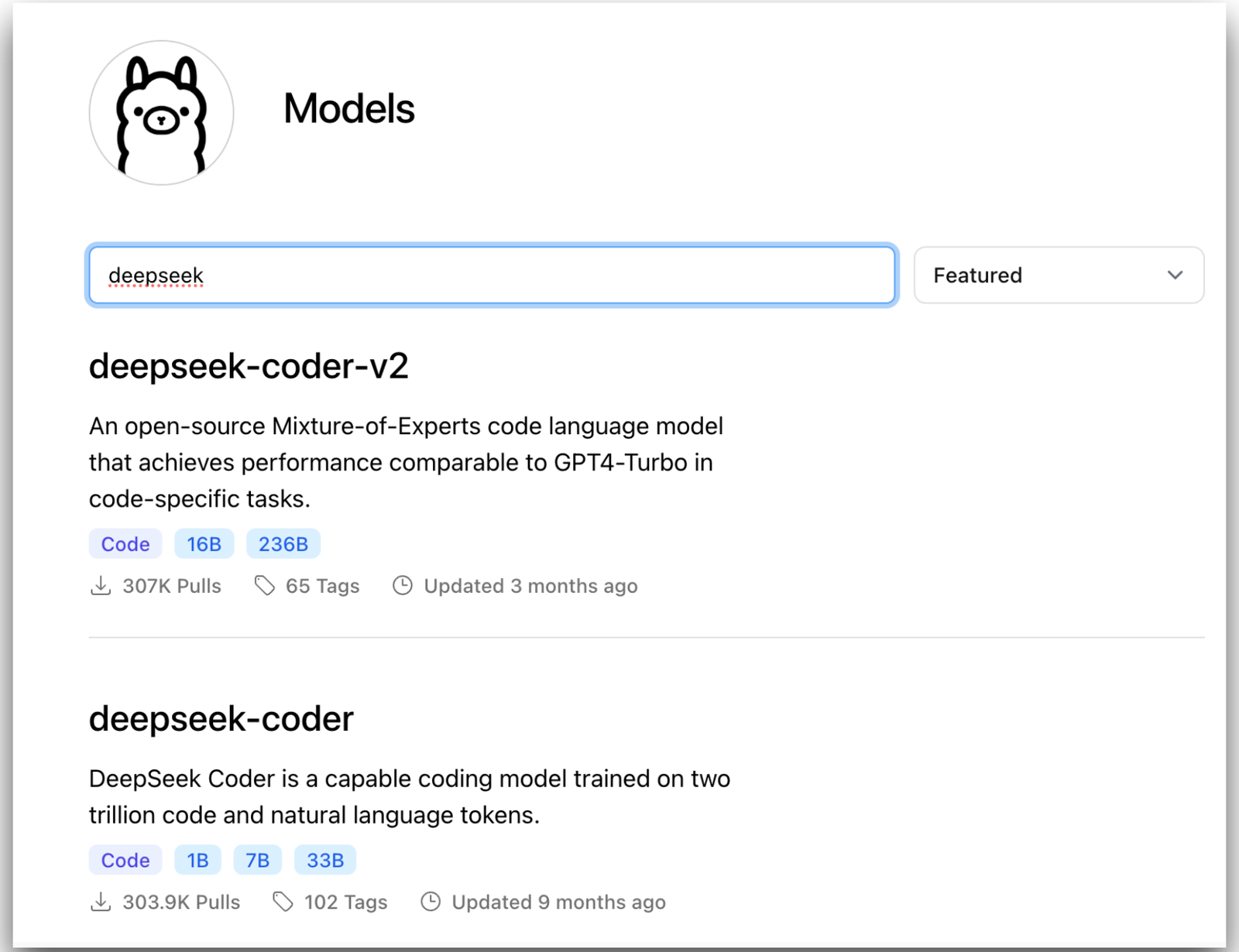
⏚ Filter model types

(•) 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	68.04
🔶	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	65.8	59.66
🔶 EXT	OpenCodeInterpreter-DS-6.7B	49.67	73.2	51.41	63.85	60.01

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

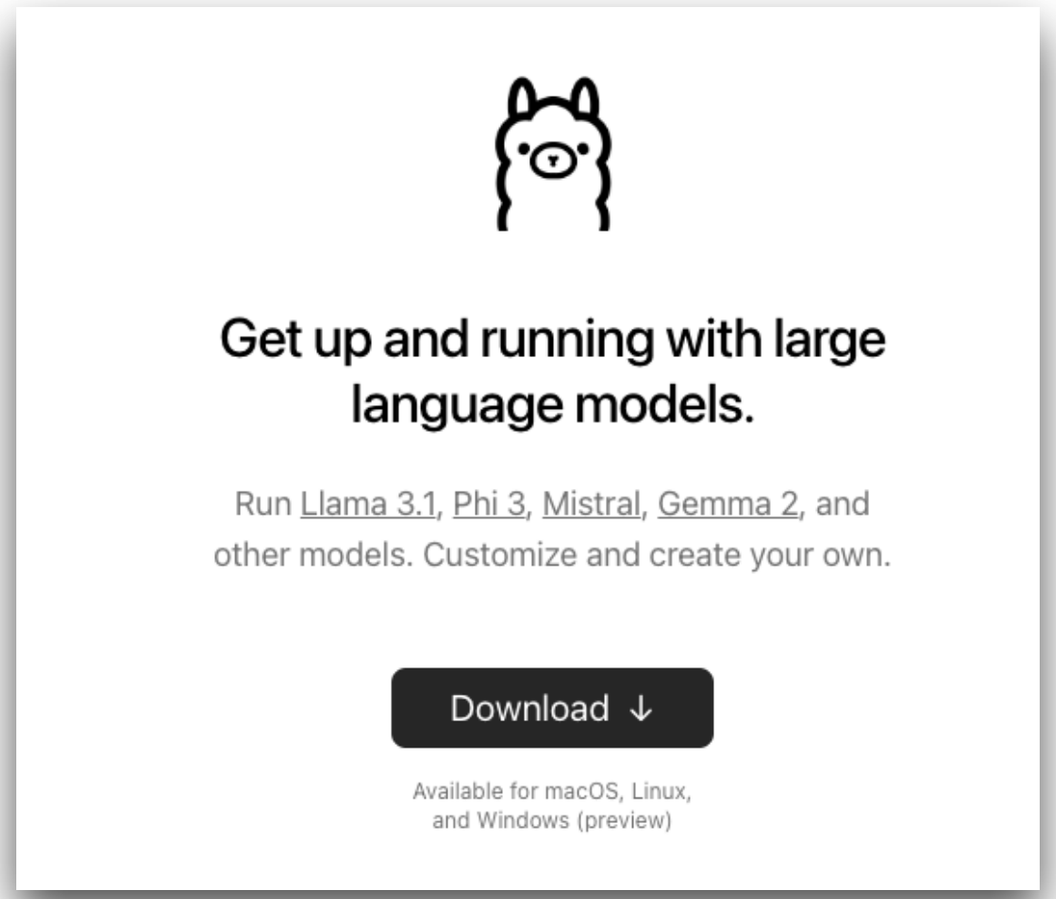
Model in Ollama



https://ollama.com/library

Workshop with Ollama

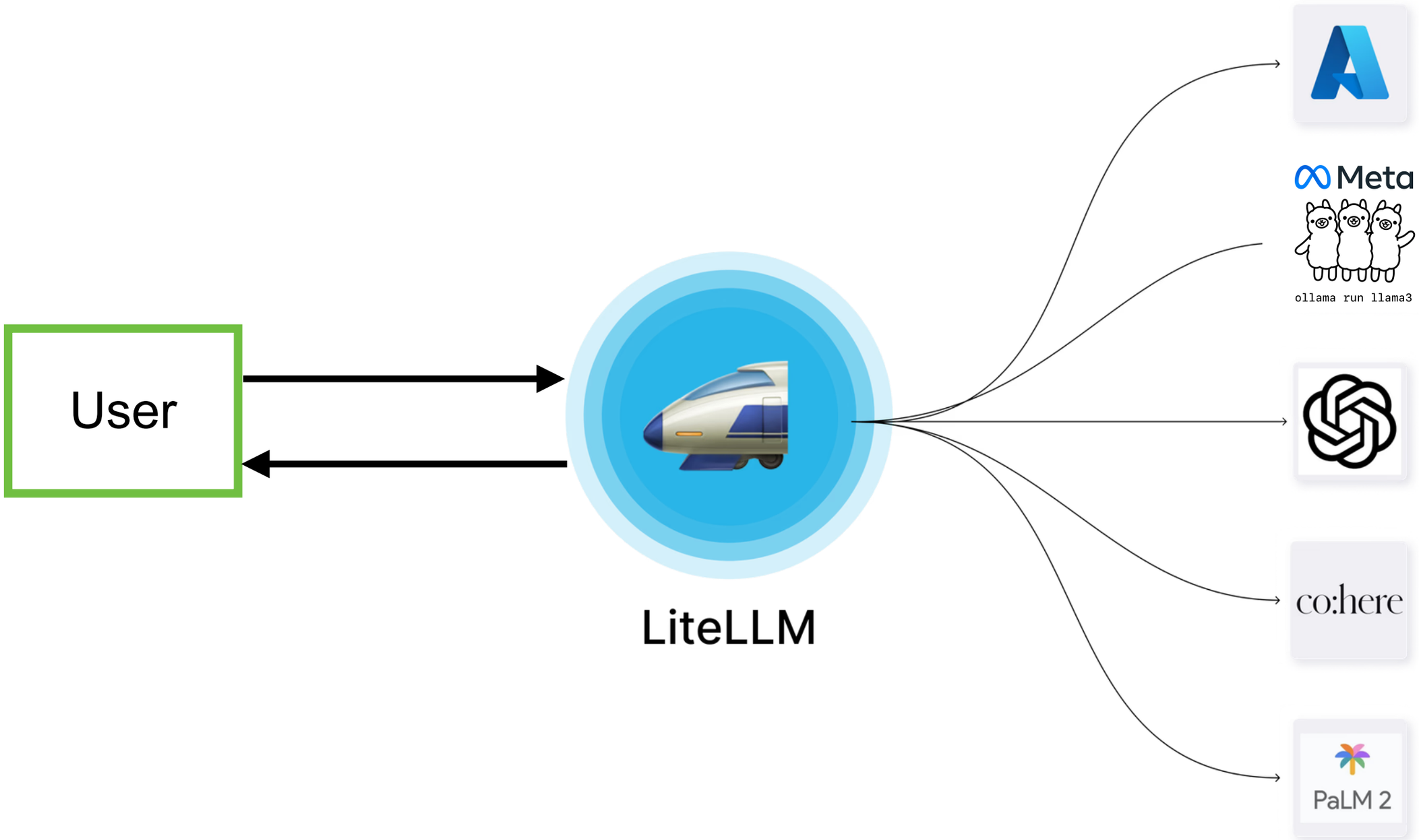
$ollama run **llama3.1**



https://github.com/up1/workshop-ai-with-technical-team/wiki/Local-LLM-with-Ollama

LiteLLM as a Proxy

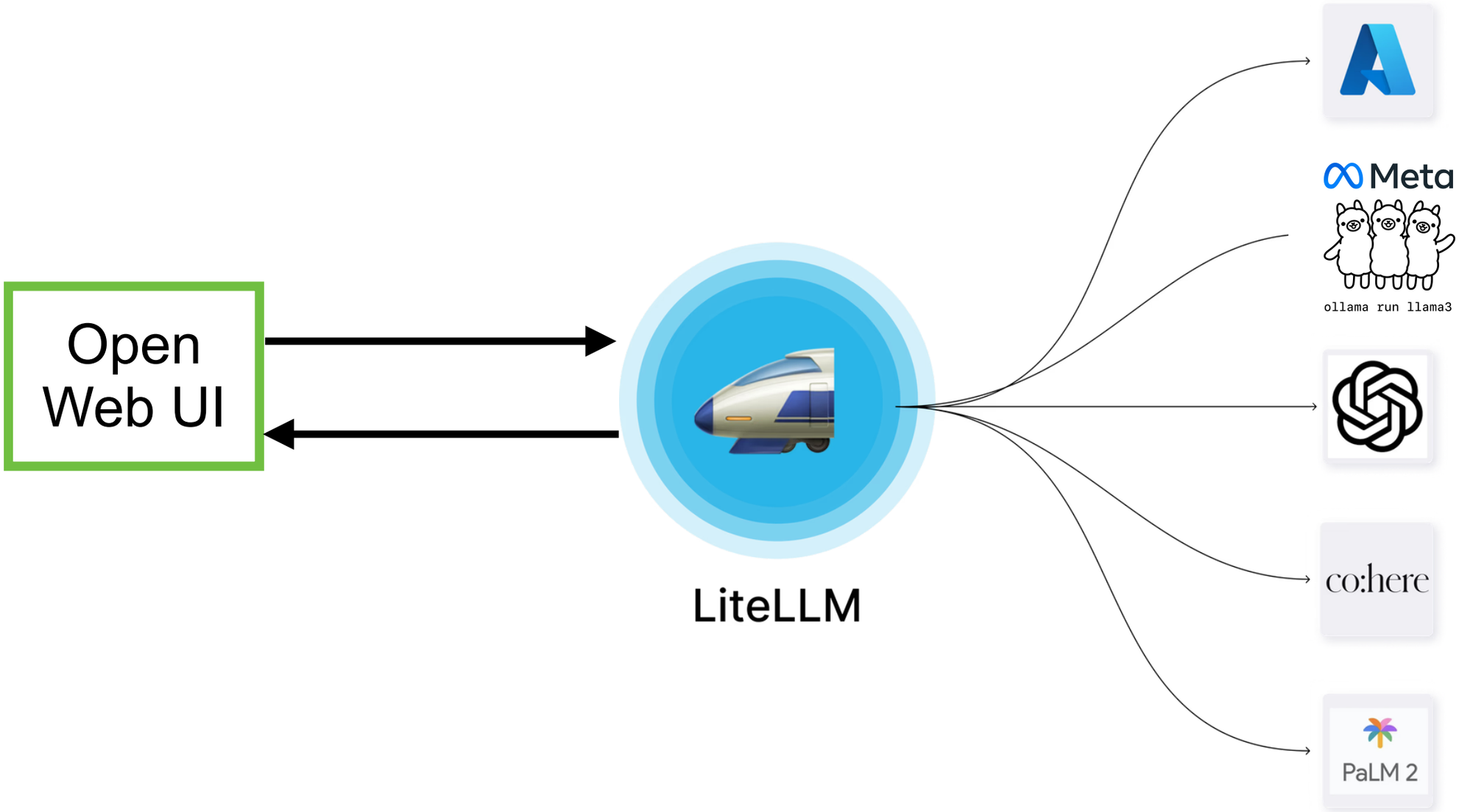
LiteLLM as a Proxy



https://www.litellm.ai/

Workshop

Use docker compose to build and run

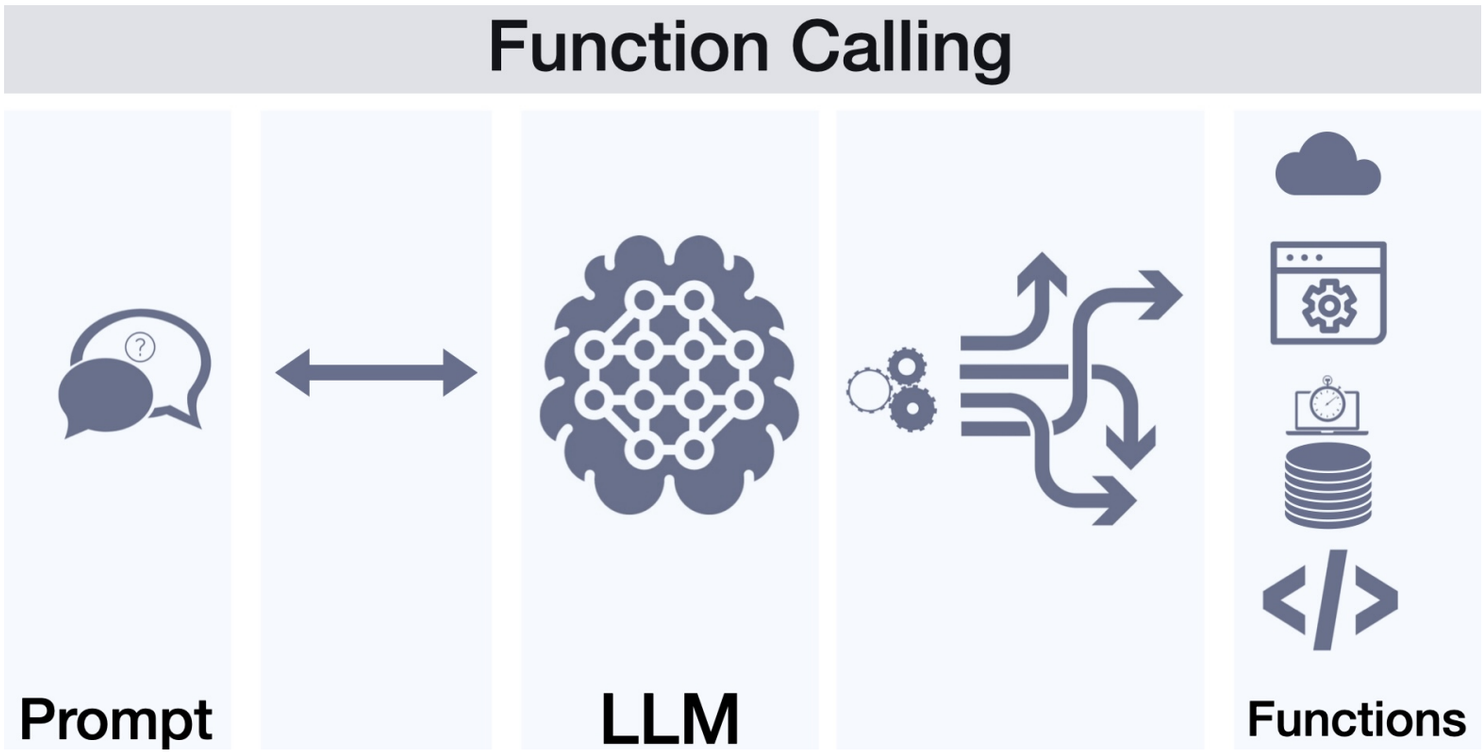


https://github.com/up1/workshop-ai-with-technical-team/wiki/LiteLLM-and-WebUI

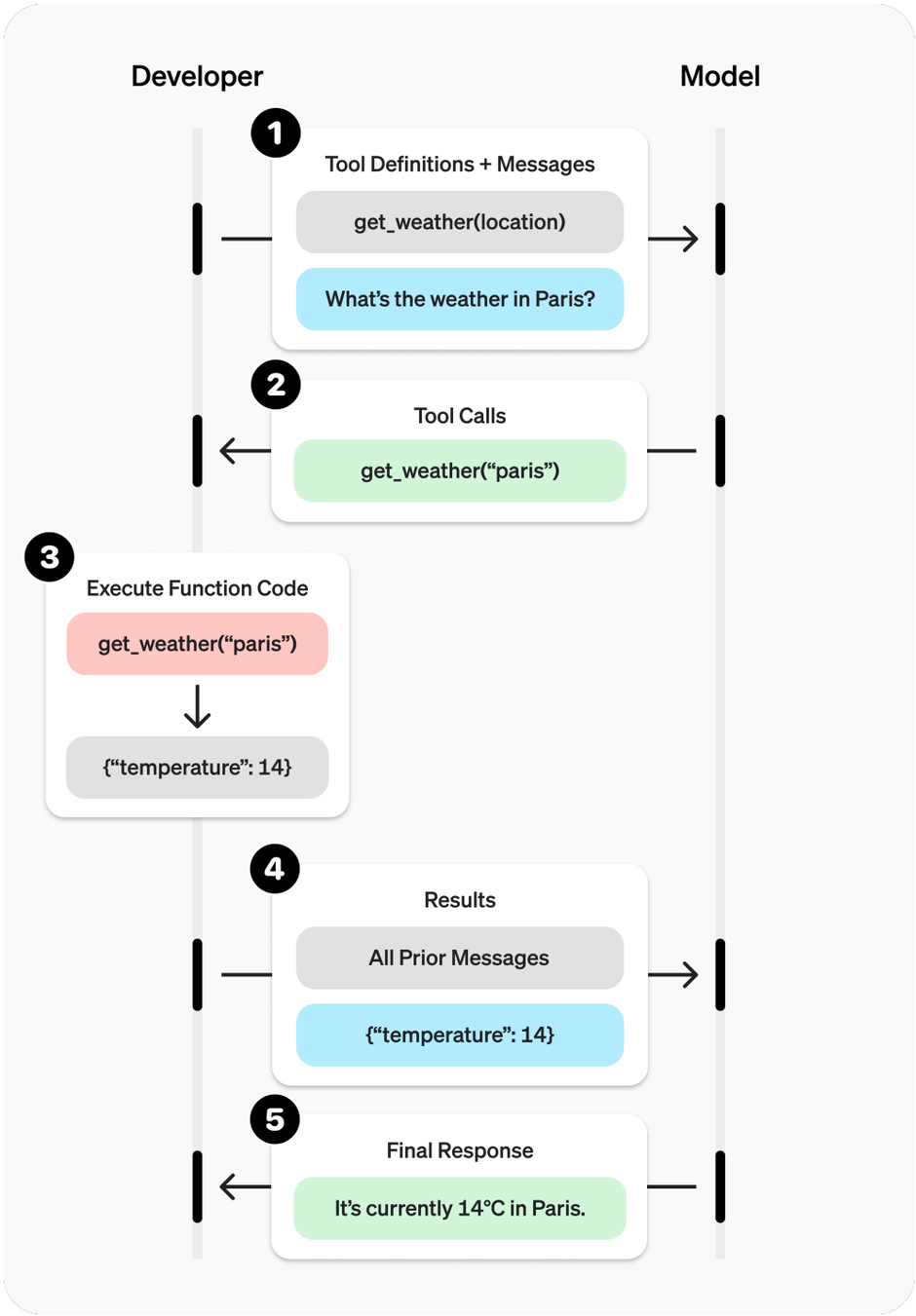
Function Calling ?

Function Calling ?

Allow LLM to recognize what tool it need based on user's input and when to invoke it.

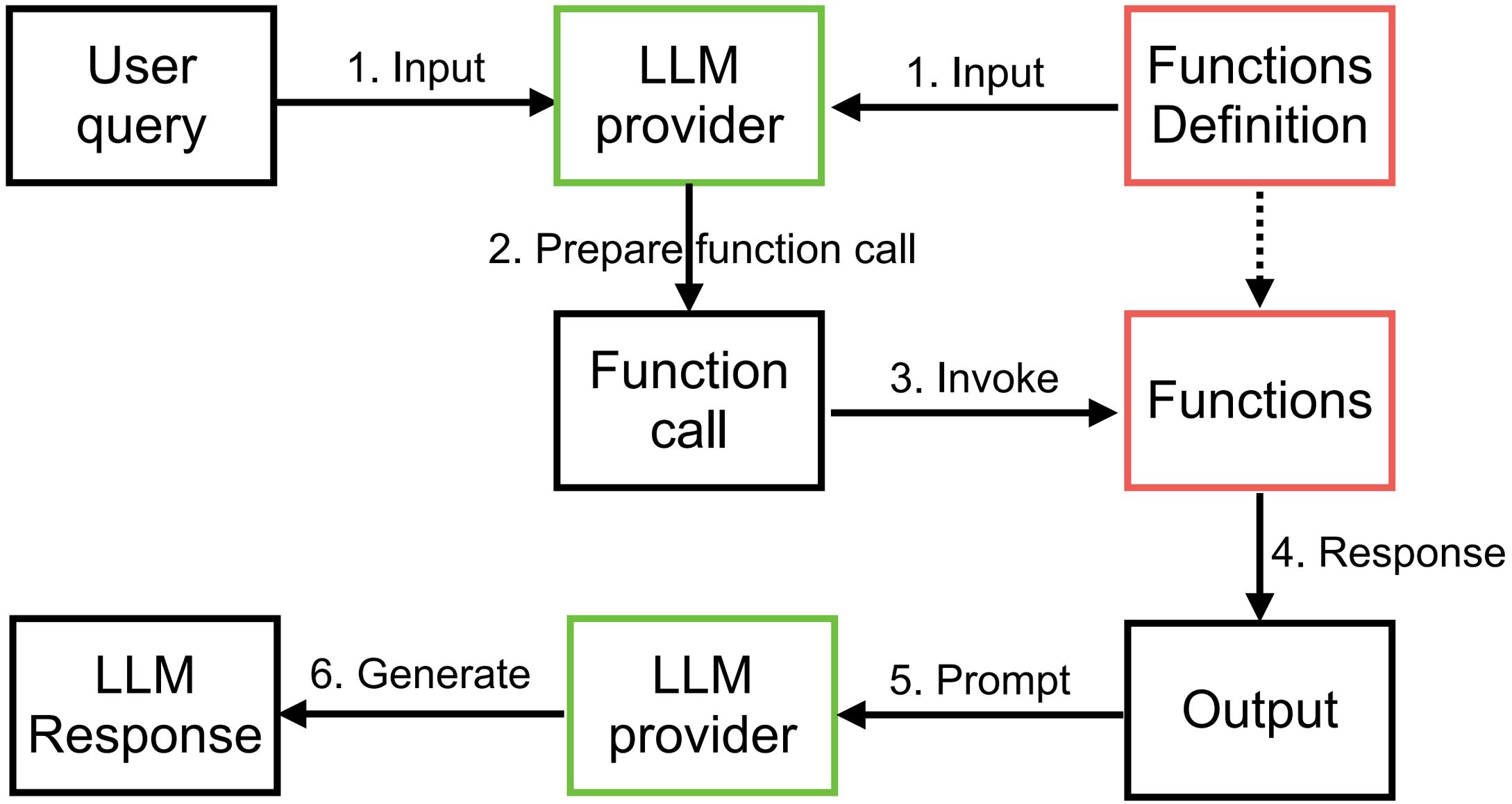


Function calling ?



https://platform.openai.com/docs/guides/function-calling?api-mode=responses

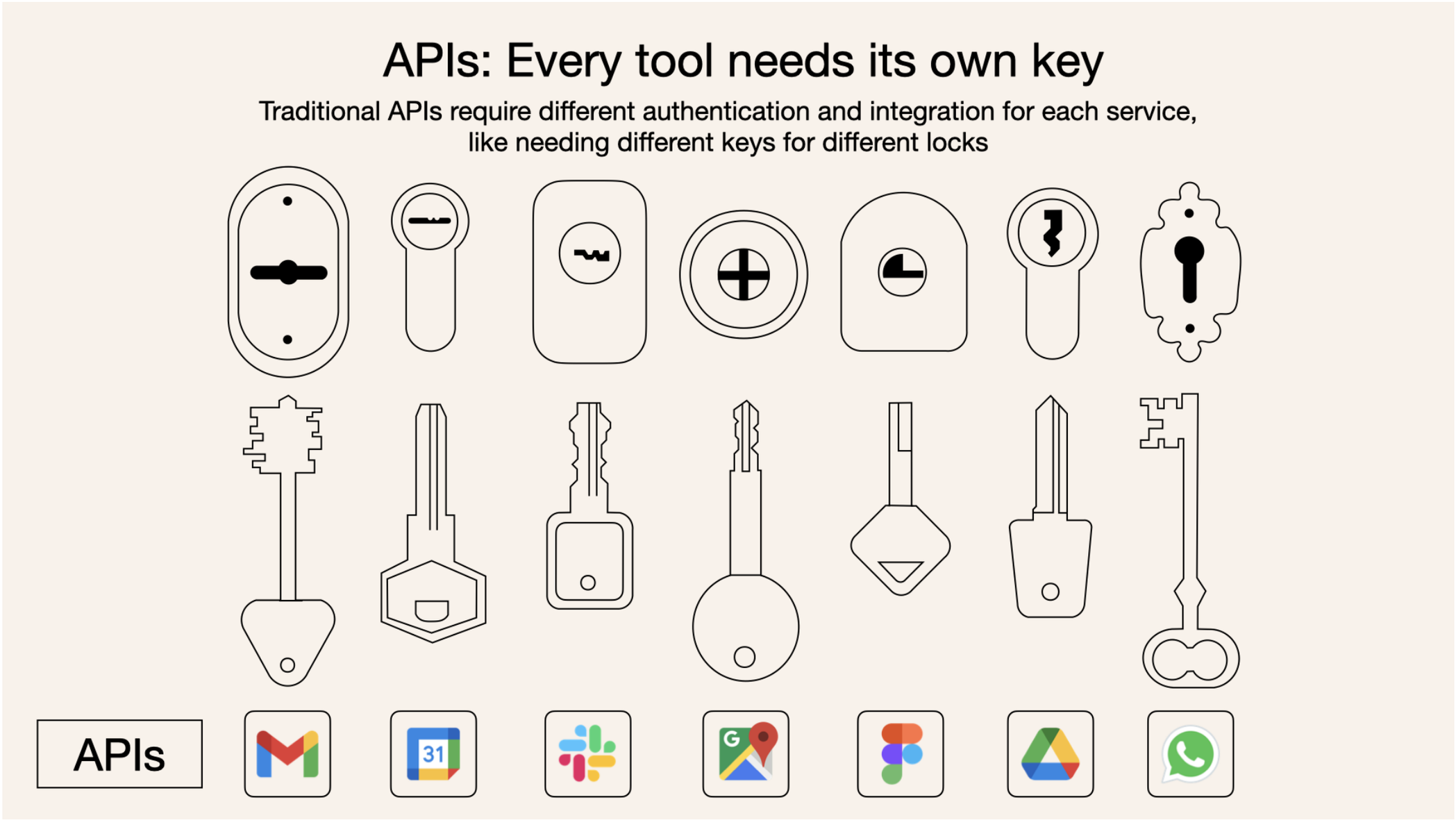
Function Calling



Support function calling

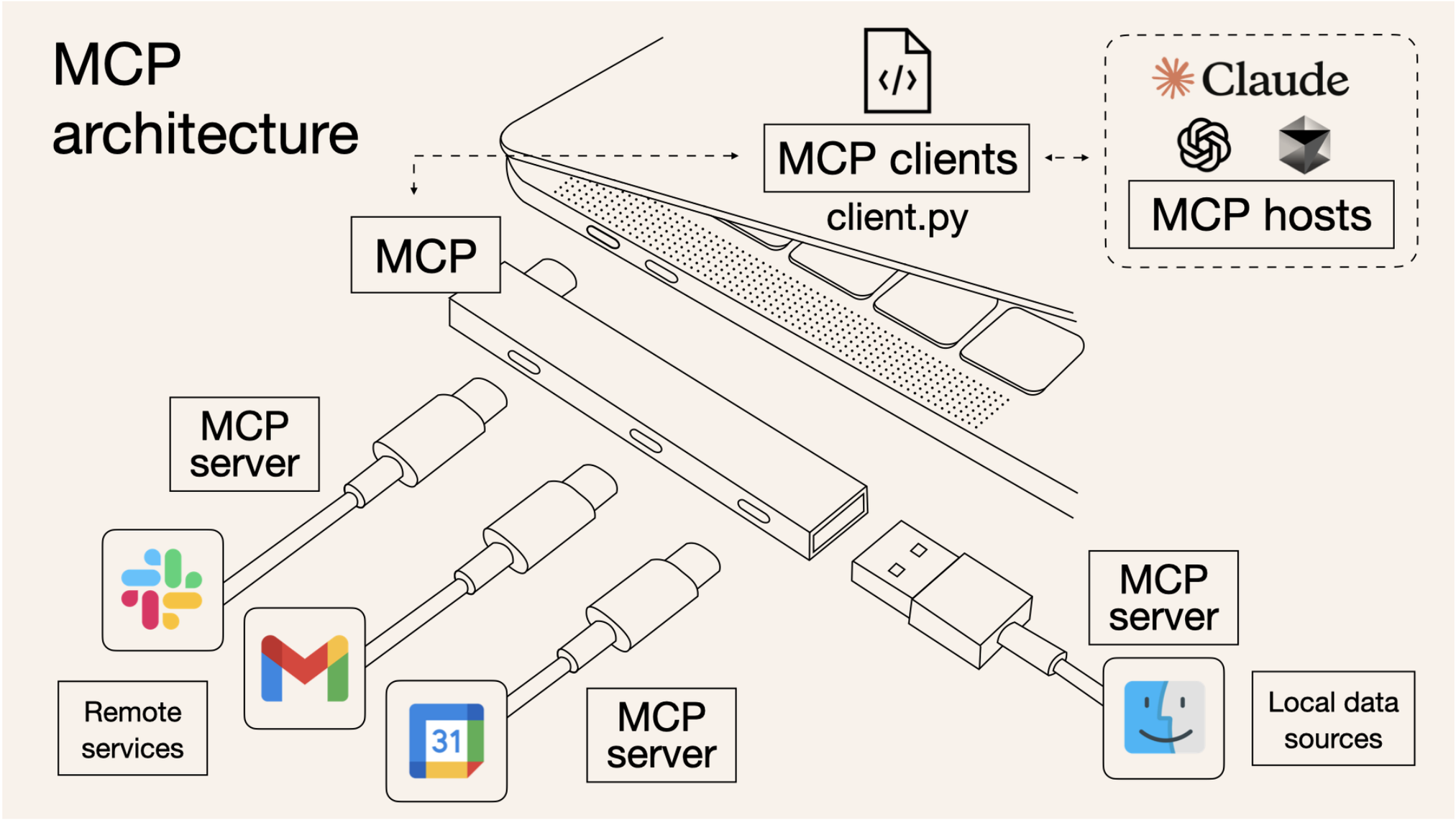
Provider name	Model name
OpenAI	GPT 4
Anthropic	Claude 3 (Sonnet, Haiku, Opus)
Google	Gemini

Function calling !!



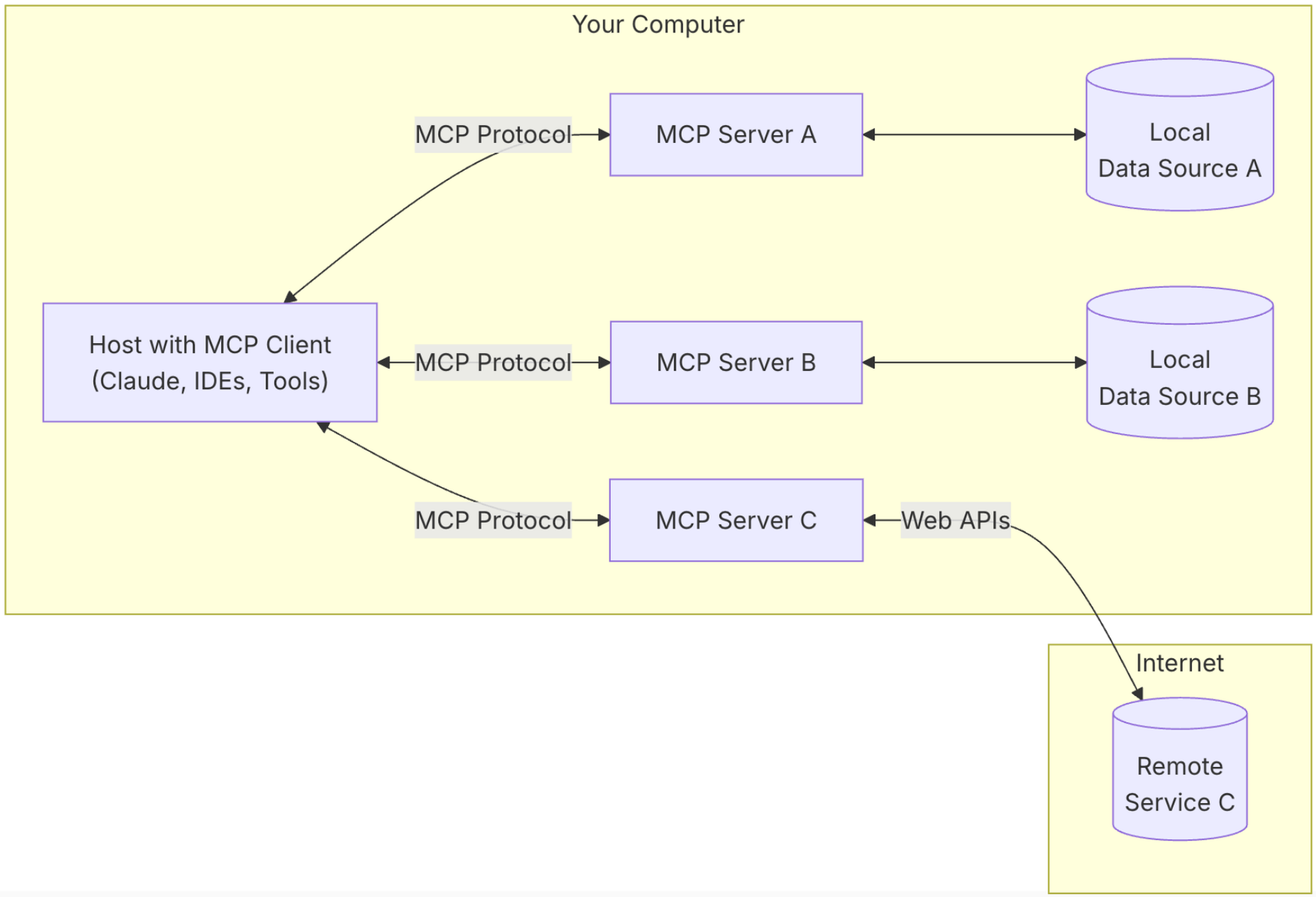
https://norahsakal.com/blog/mcp-vs-api-model-context-protocol-explained/

Standardized function calling !!



https://norahsakal.com/blog/mcp-vs-api-model-context-protocol-explained/

Model Context Protocol



https://modelcontextprotocol.io/

Retrieval-Augmented Generation (RAG)

RAG

Enhances LLMs by retrieving external knowledge before generating response

Improve accuracy	Reduce hallucinations	Real-time knowledge updates

Core Components

Retriever

Fetch relevant documents from a knowledge base

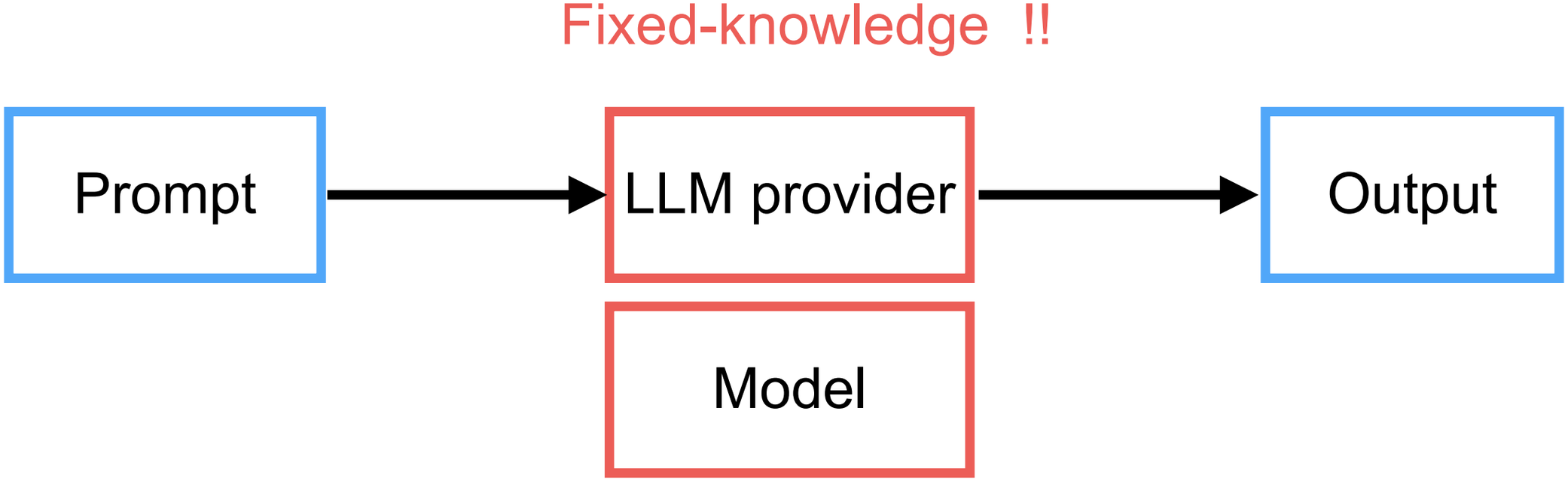
Generator

Uses the retrieved information to generate a response

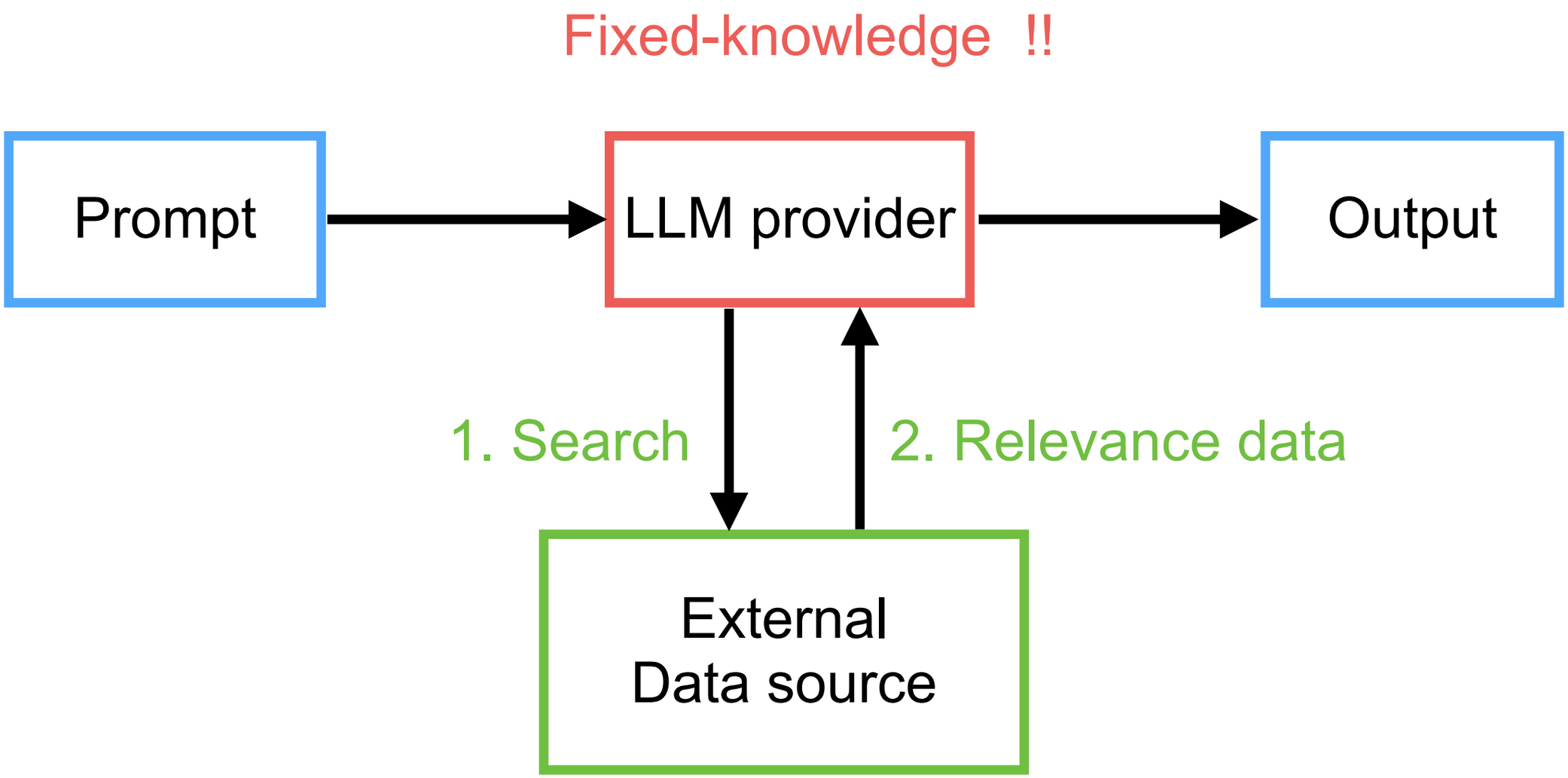
Enhancements

Different RAG architectures modify how retrieval and generation interact to improve performance

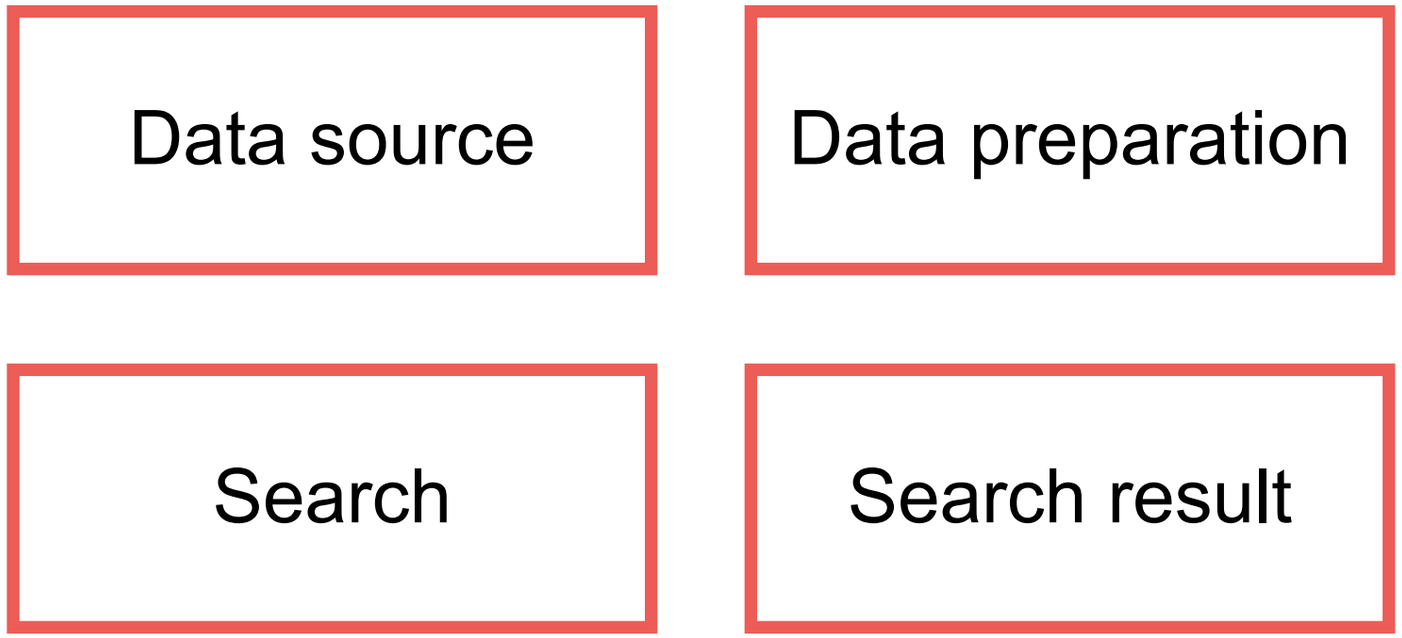
Limitation of LLM



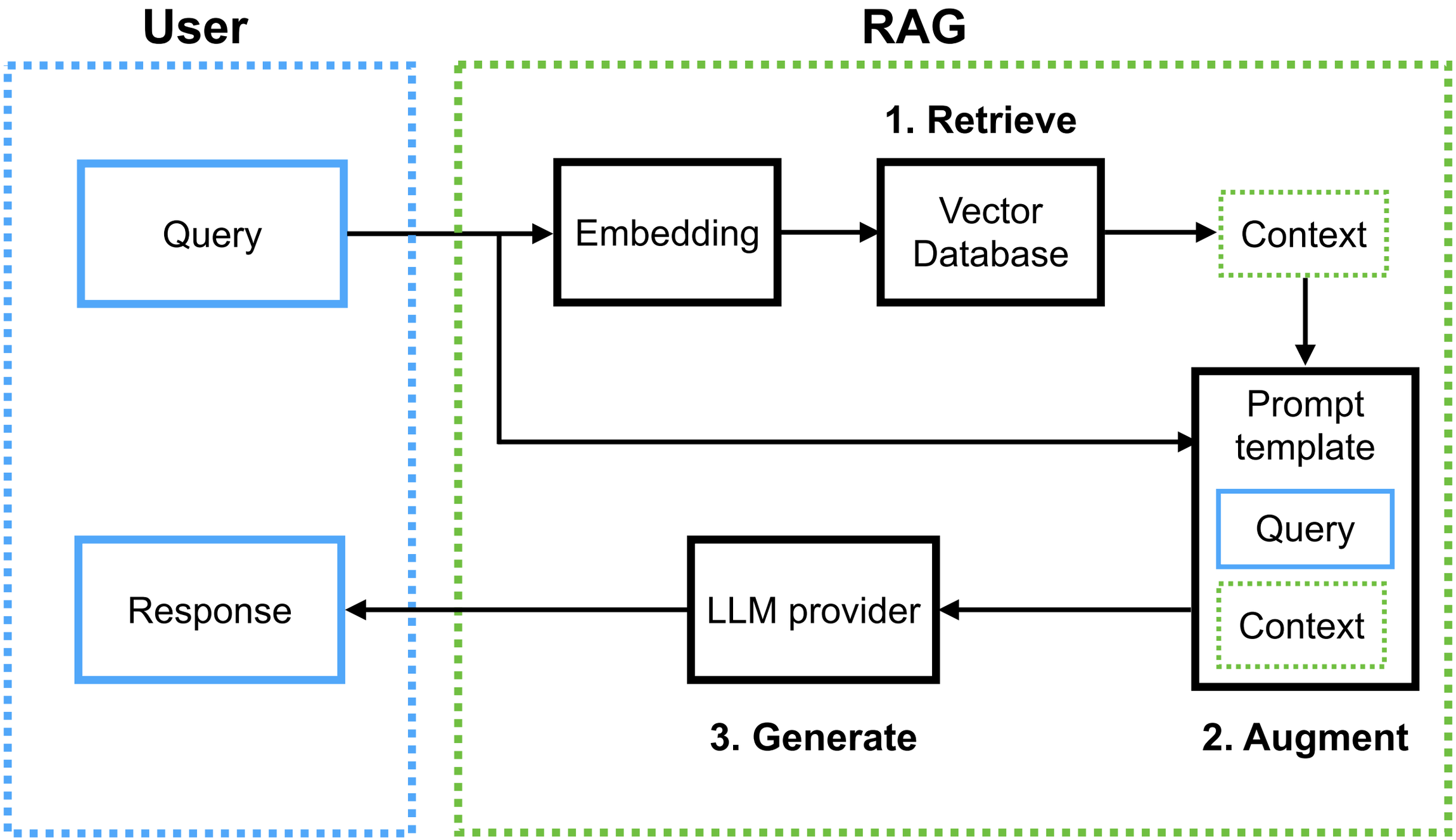
RAG ?



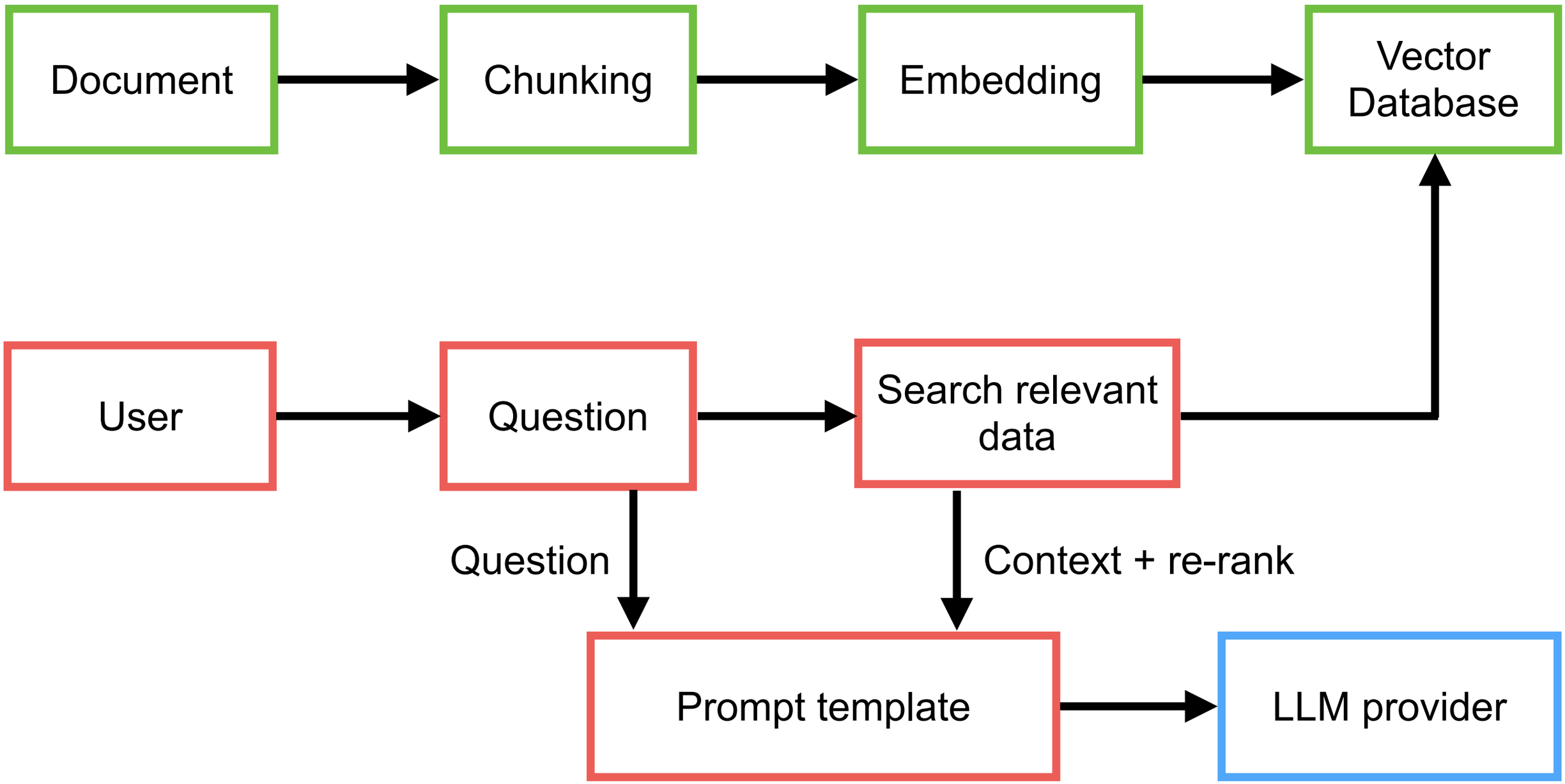
How to search/retrieve data ?



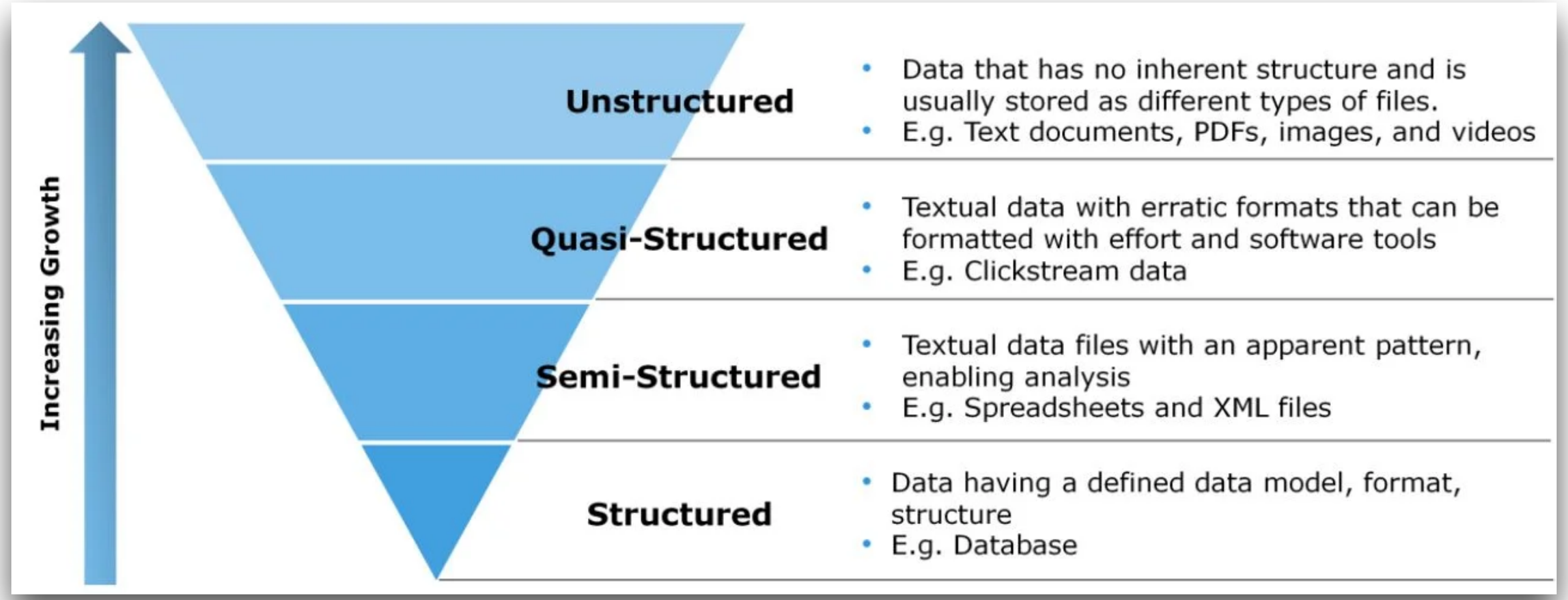
Basic RAG Architecture



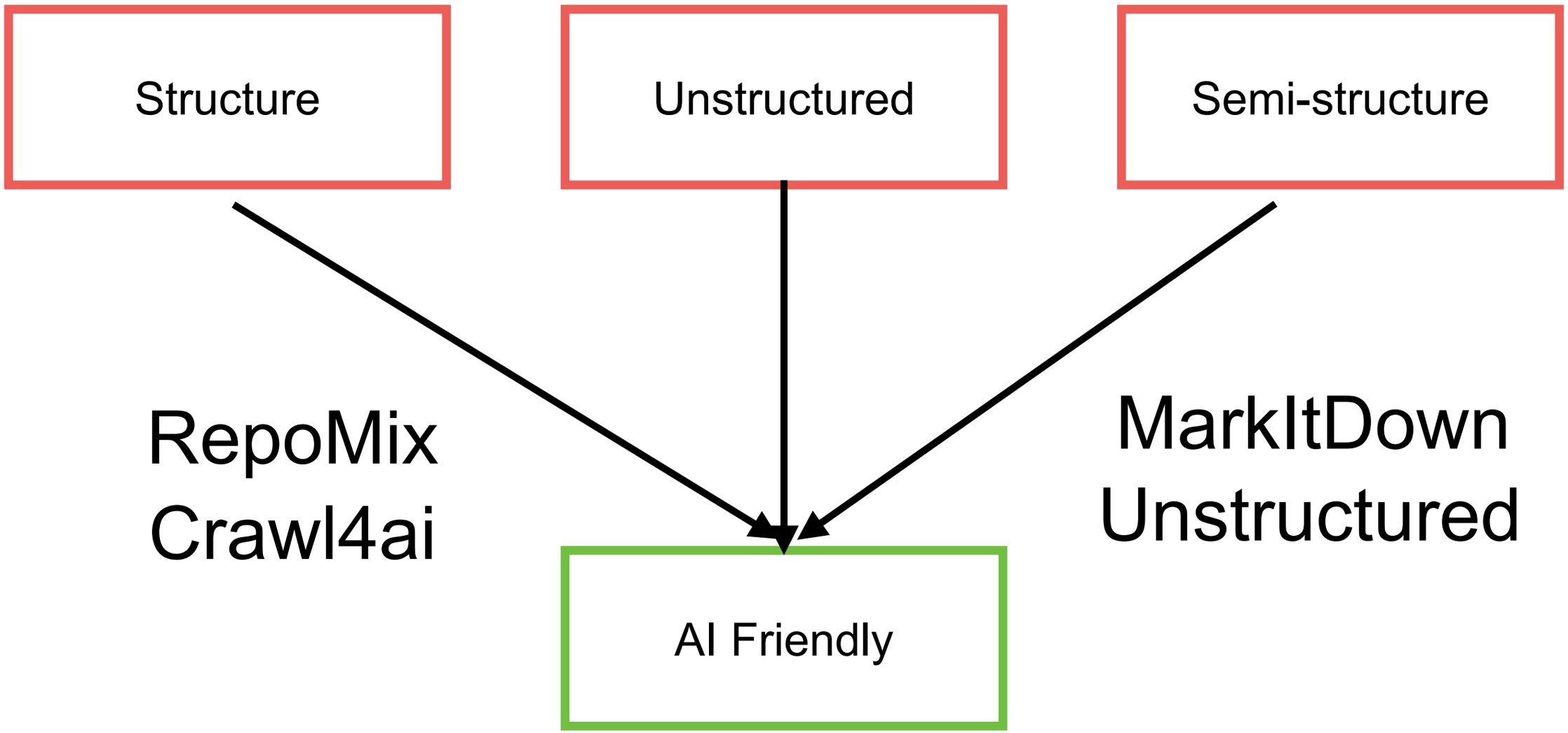
RAG process



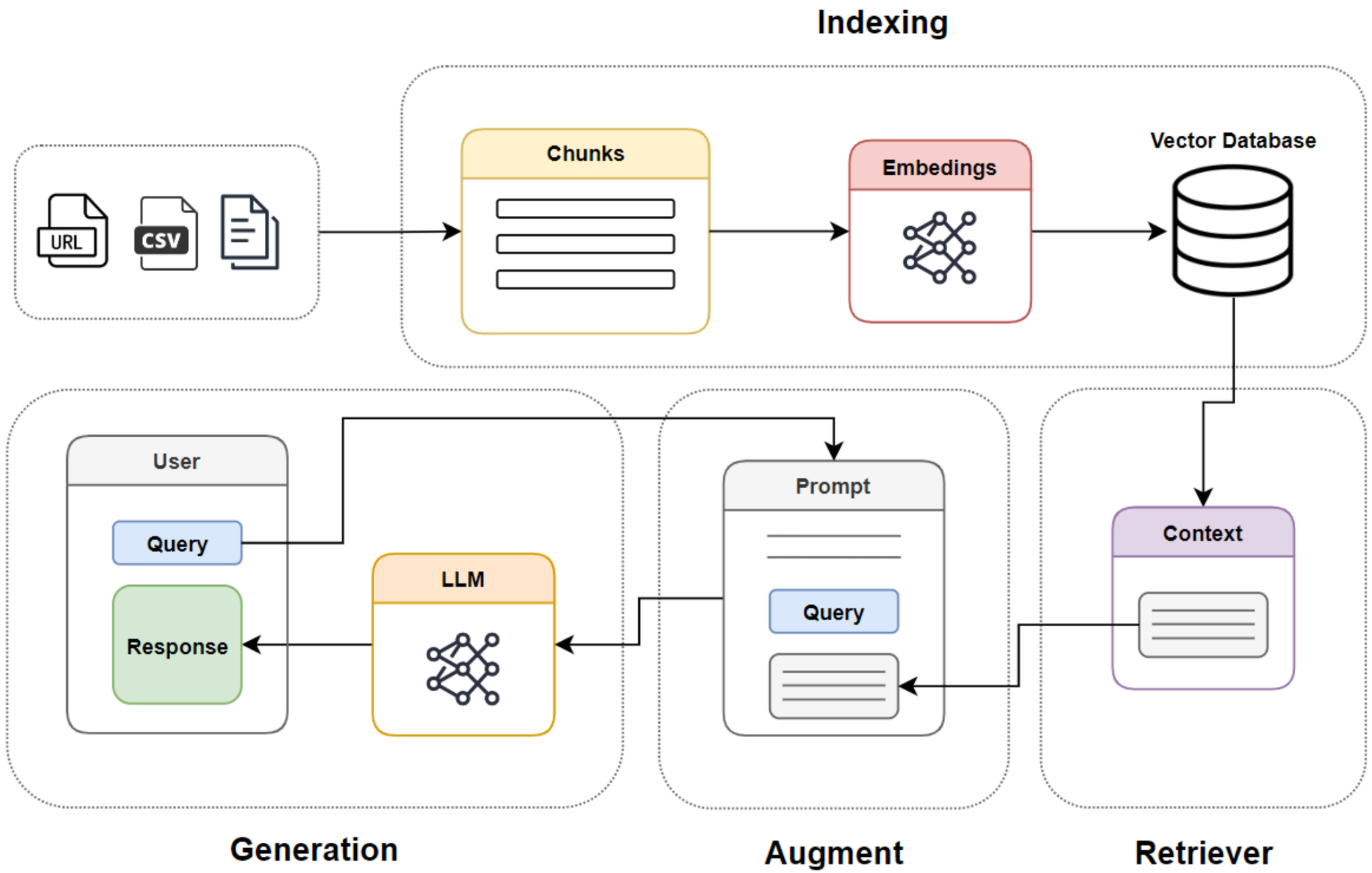
Structures of Data ?



Friendly Data for LLM/AI

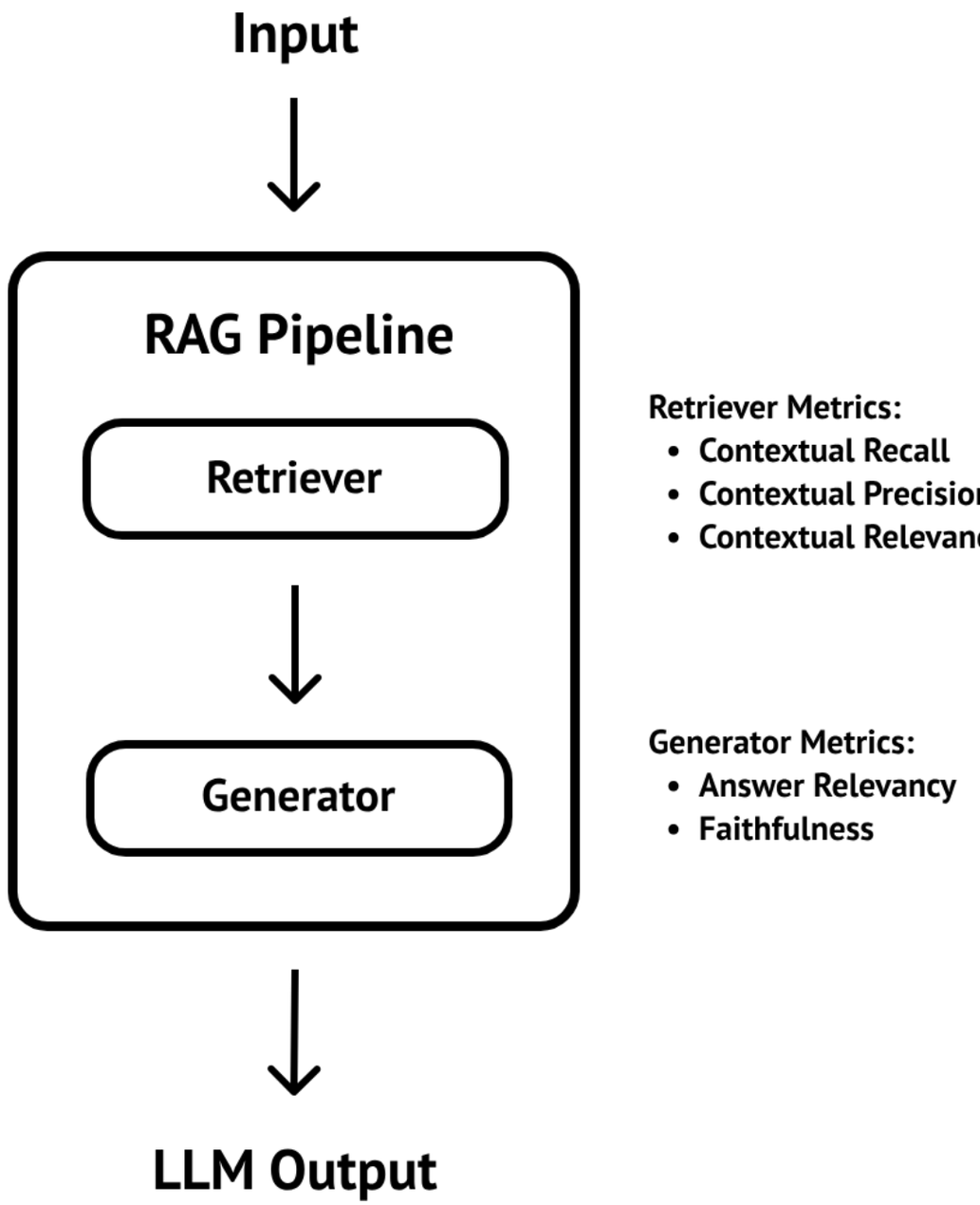


RAG Cookbook (techniques)

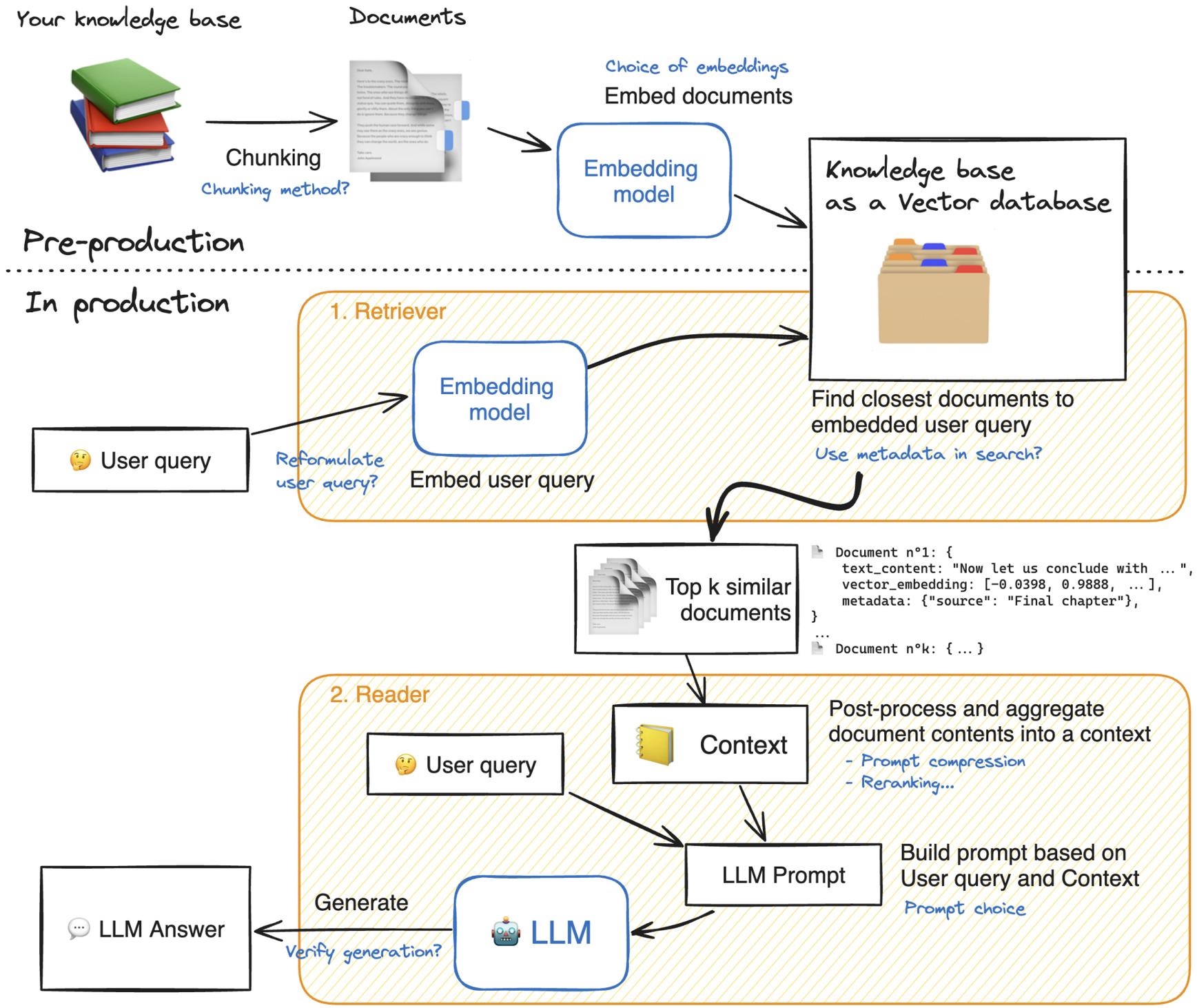


https://github.com/athina-ai/rag-cookbooks

RAG Evaluation !!

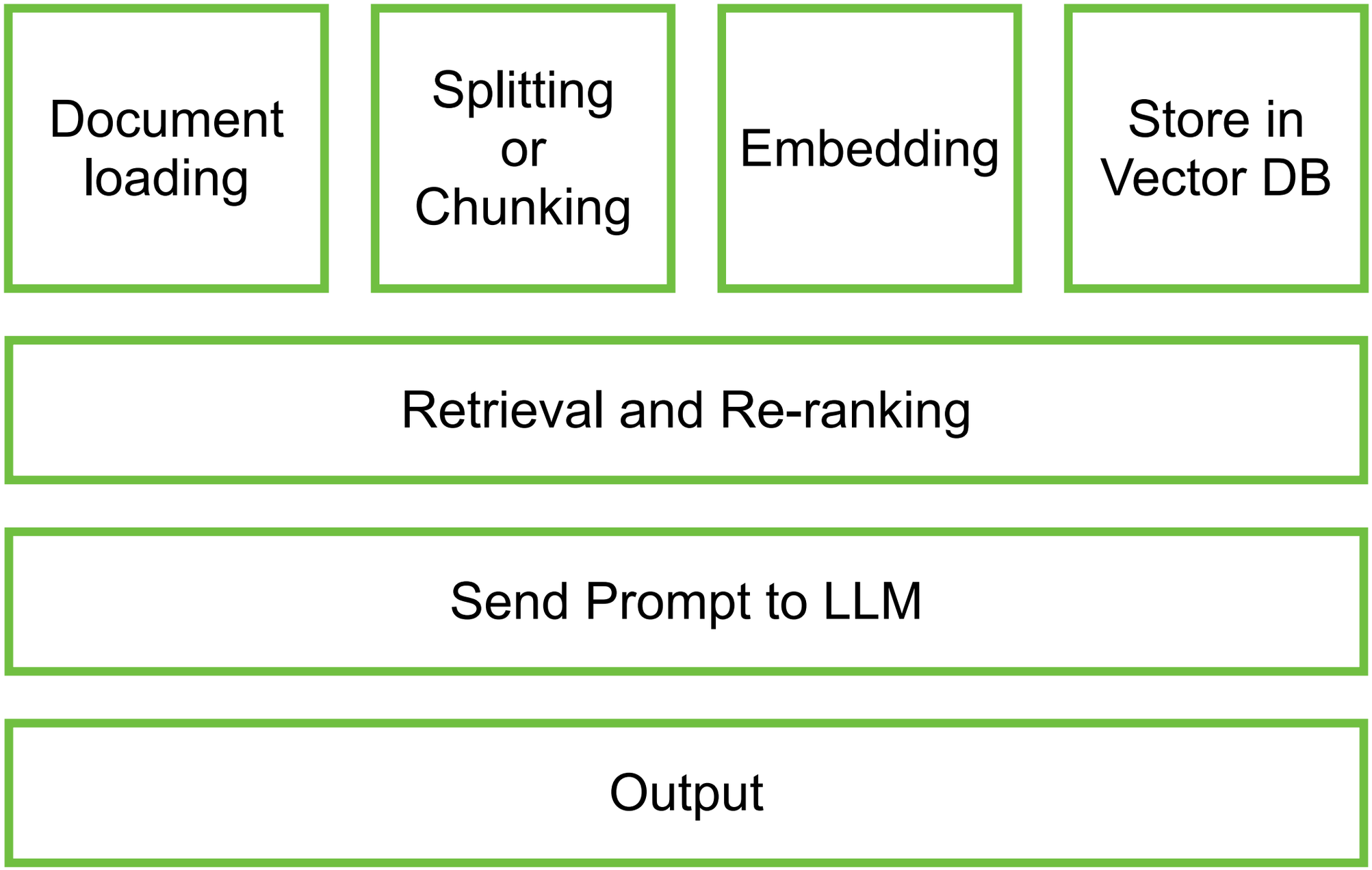


https://www.deepeval.com/guides/guides-rag-evaluation



https://huggingface.co/learn/cookbook/en/rag_evaluation

RAG Implementation



Failure Points of RAG

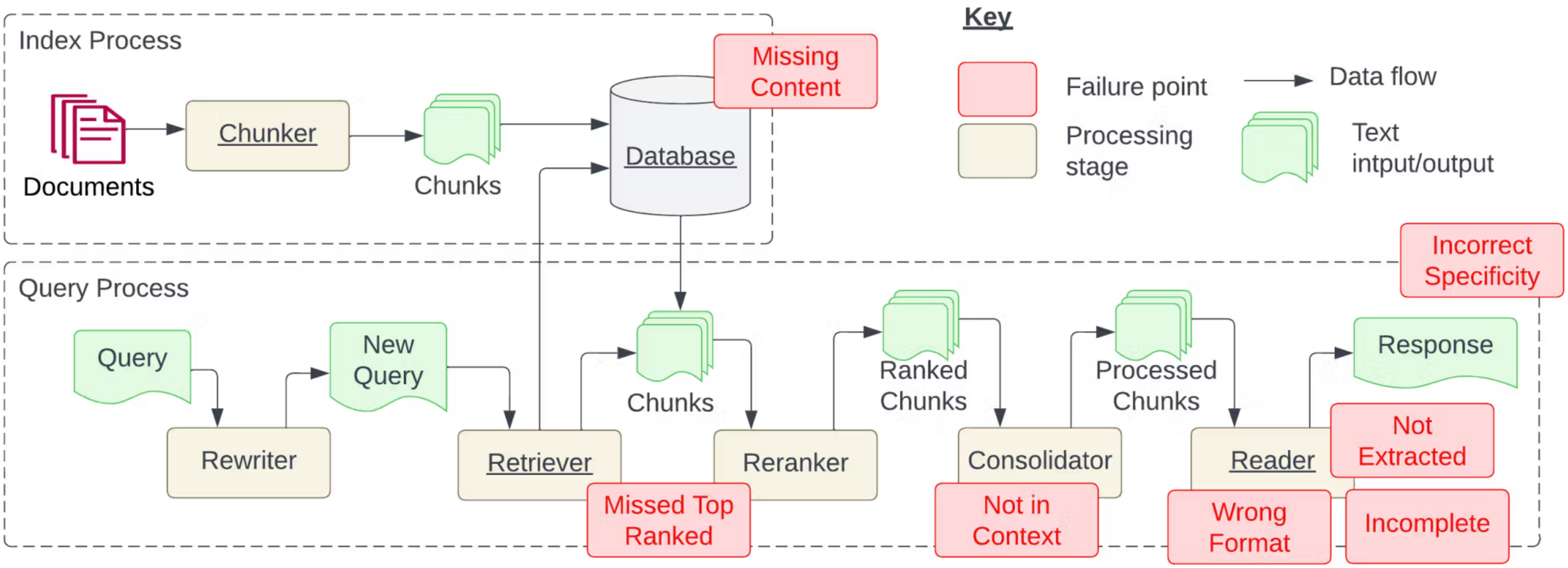


Figure 1: Indexing and Query processes required for creating a Retrieval Augmented Generation (RAG) system. The indexing process is typically done at development time and queries at runtime. Failure points identified in this study are shown in red boxes. All required stages are underlined. Figure expanded from [19].

https://www.galileo.ai/blog/mastering-rag-how-to-architect-an-enterprise-rag-system

RAG is better

But come with Cost !!

More latency

Retrieval errors

Accuracy !!

More Complexity

Maintenance overhead

RAG Techniques ?

Semantic chunking

Chunk size selector

Context chunk header

Adaptive RAG

Re-ranking

Graph TAG

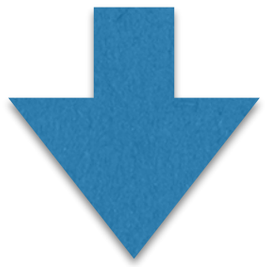
https://github.com/FareedKhan-dev/all-rag-techniques

Pre-Retrieval optimization ?

Preprocessing and cleansing data
Chunking strategies
Add metadata
Embedding model selection

Chunking strategies

Size of Document
> context window size



Chunking

Chunk 1

Chunk 2

Chunk 3

Chunking Strategies !!

Fixed size
Recursive characters
Document structure-based
Semantic chunking
Agentic chunking

. . .

https://blog.dailydoseofds.com/p/5-chunking-strategies-for-rag

Good Chunking

Efficiency

Relevance

Context preservation

Improve content generation

Reduce noise

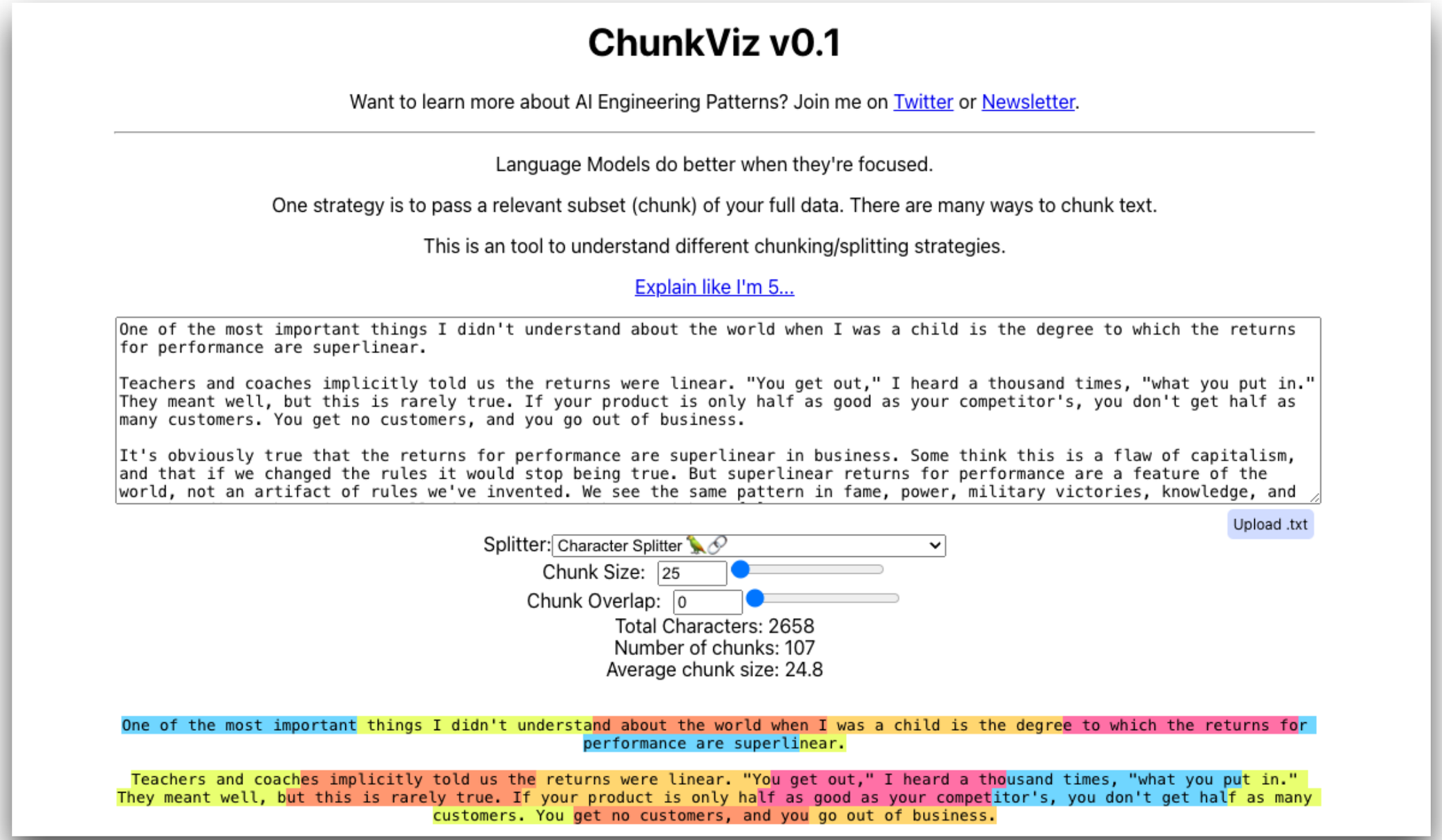
Bad Chunking

Loss context

Redundancy

Inconsistency

Chunking Visualization



https://chunkviz.up.railway.app/

Retrieval optimization ?

Re-ranking

Hybrid search

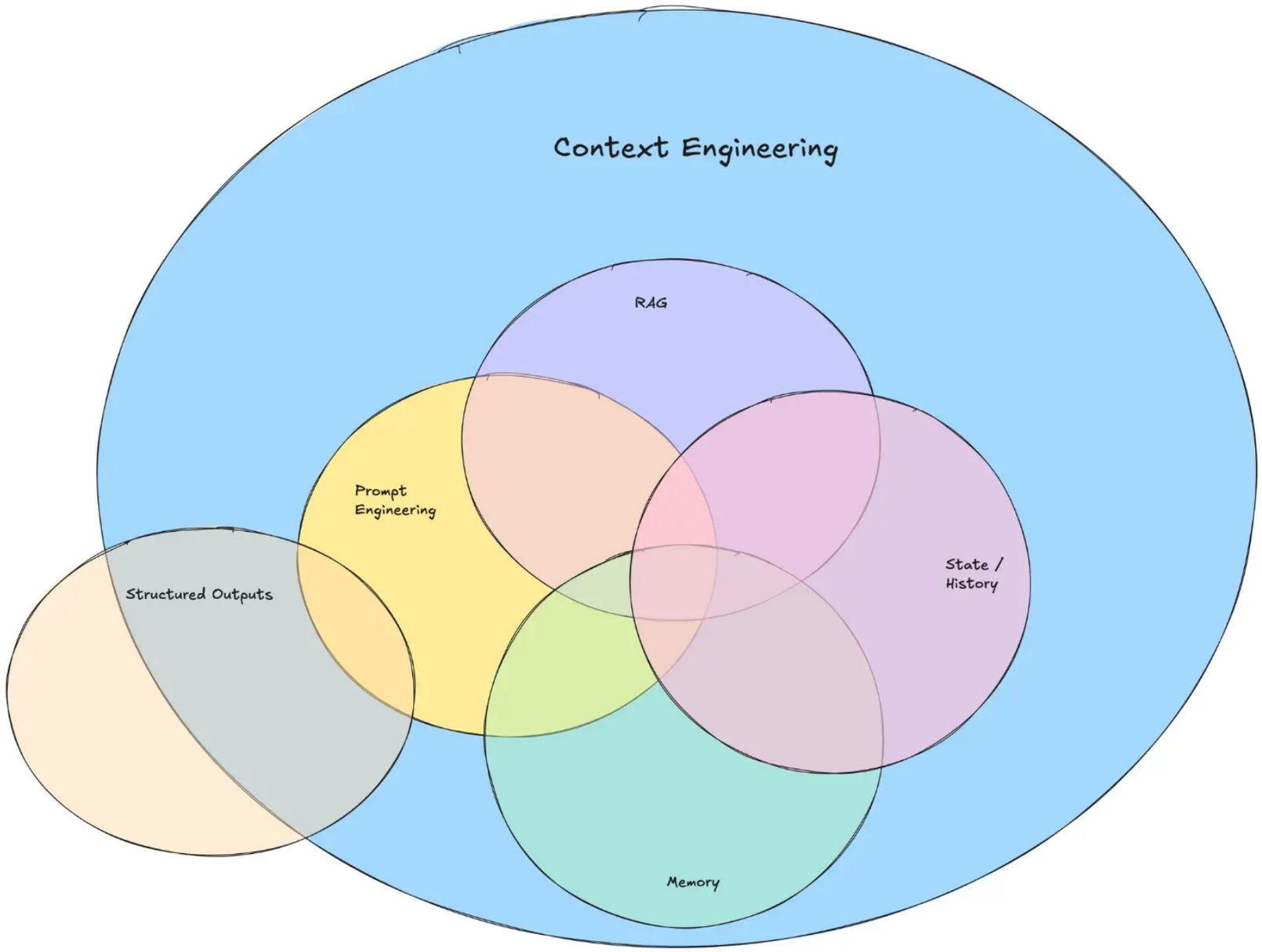
Query transformation

Multi-vector embedding

Contextual retrieval

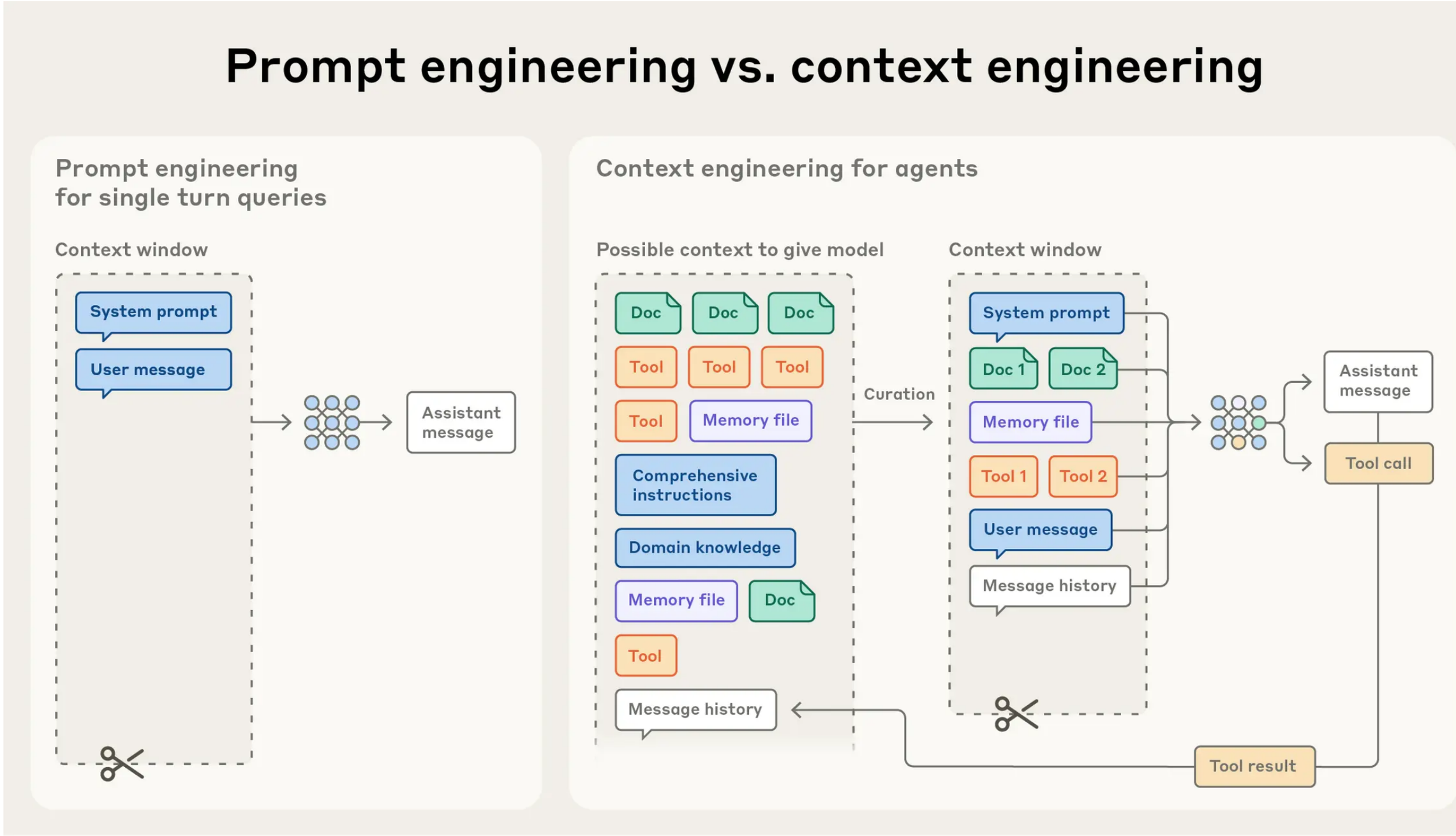
Vector database Selection

Context Engineering



https://www.promptingguide.ai/guides/context-engineering-guide

Effective Context Engineering

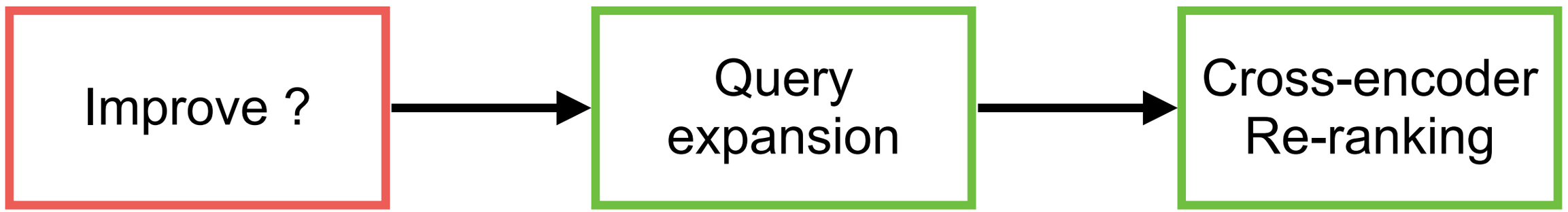


https://www.anthropic.com/engineering/effective-context-engineering-for-ai-agents

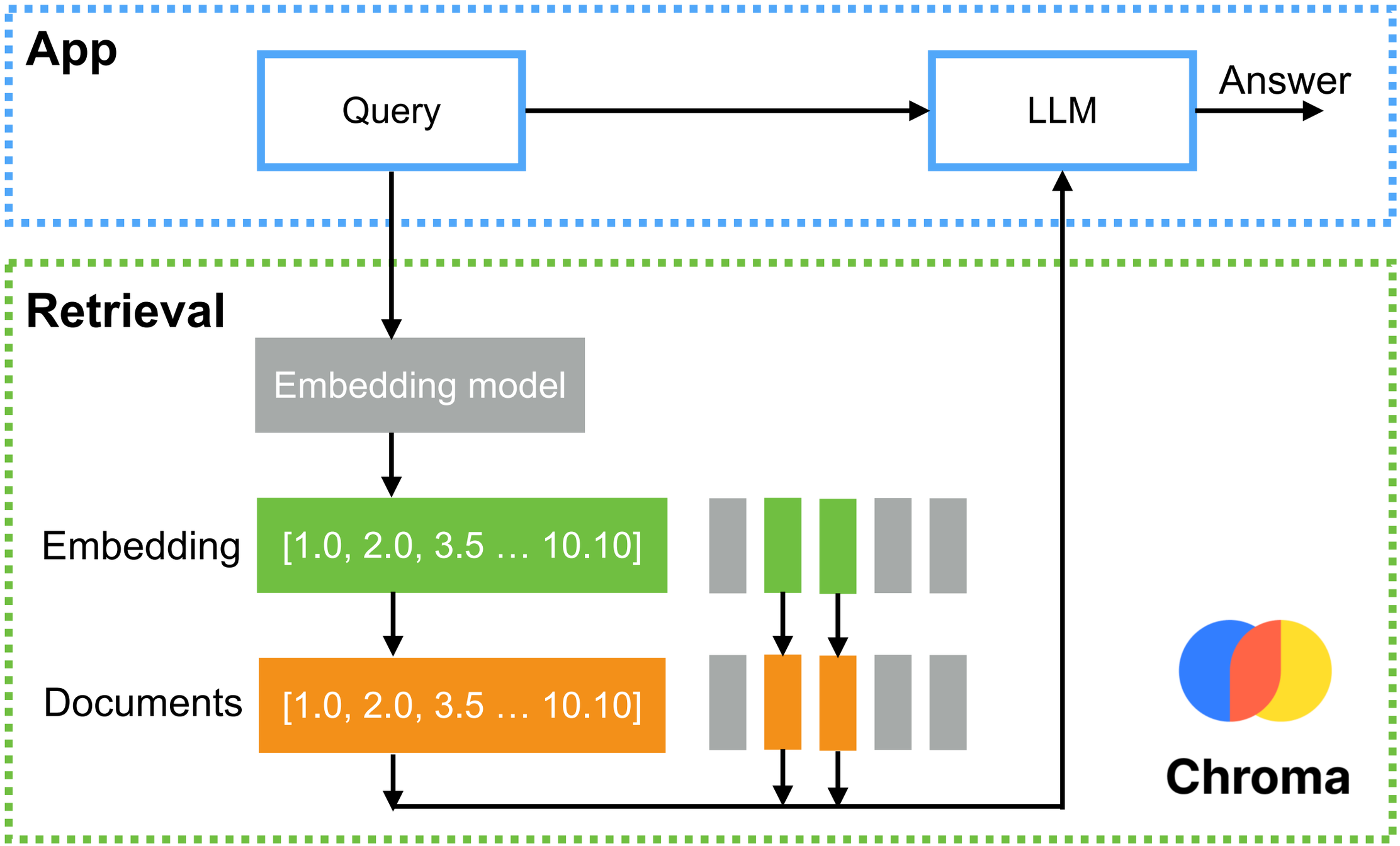
RAG Workshop

https://github.com/up1/workshop-basic-llm/tree/main/workshop/basic-rag

RAG workshop



Basic RAG



Query Expansion

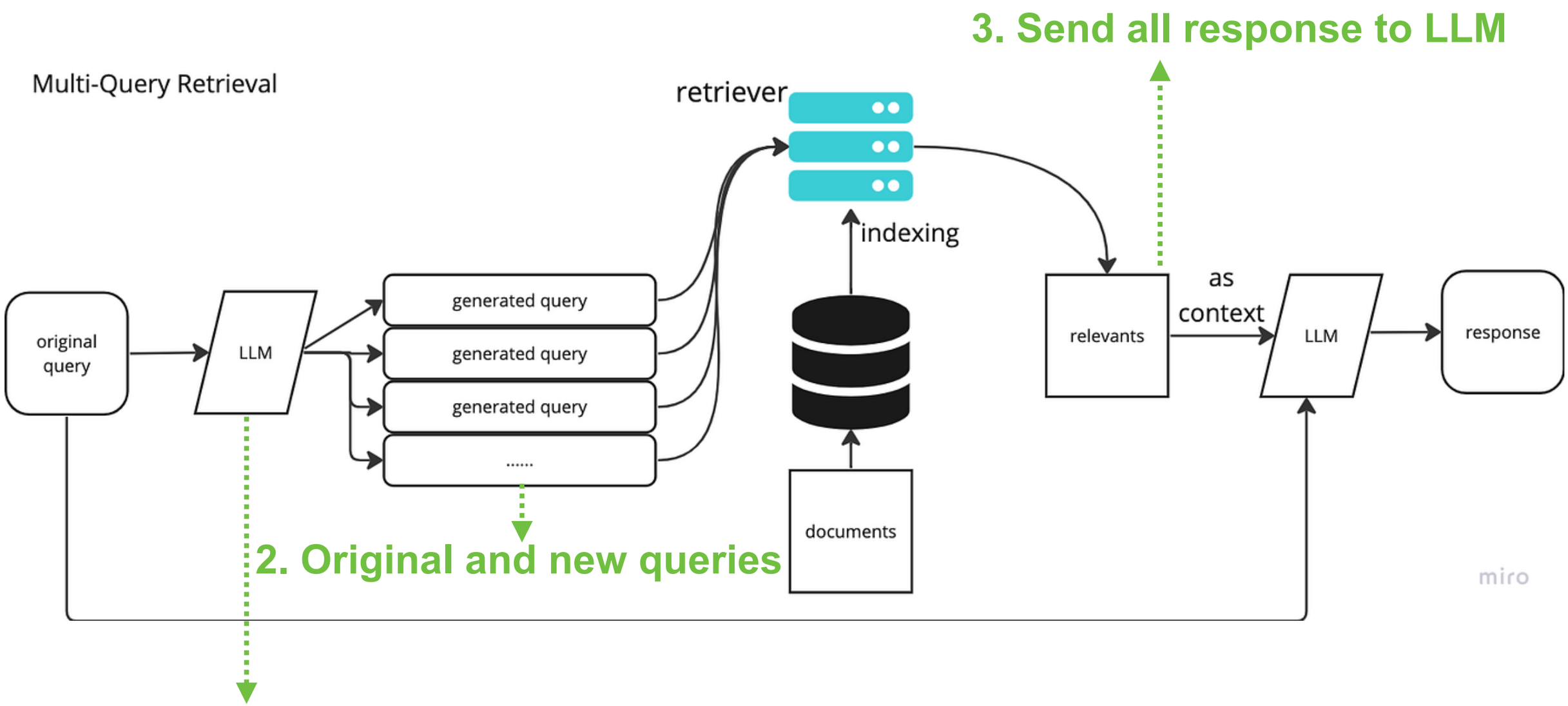
Query expansion is a widely used technique to improve the recall of search systems

Ambiguity	Vocabulary mismatch	Lack of context

"Add synonyms, related context and contextual terms"

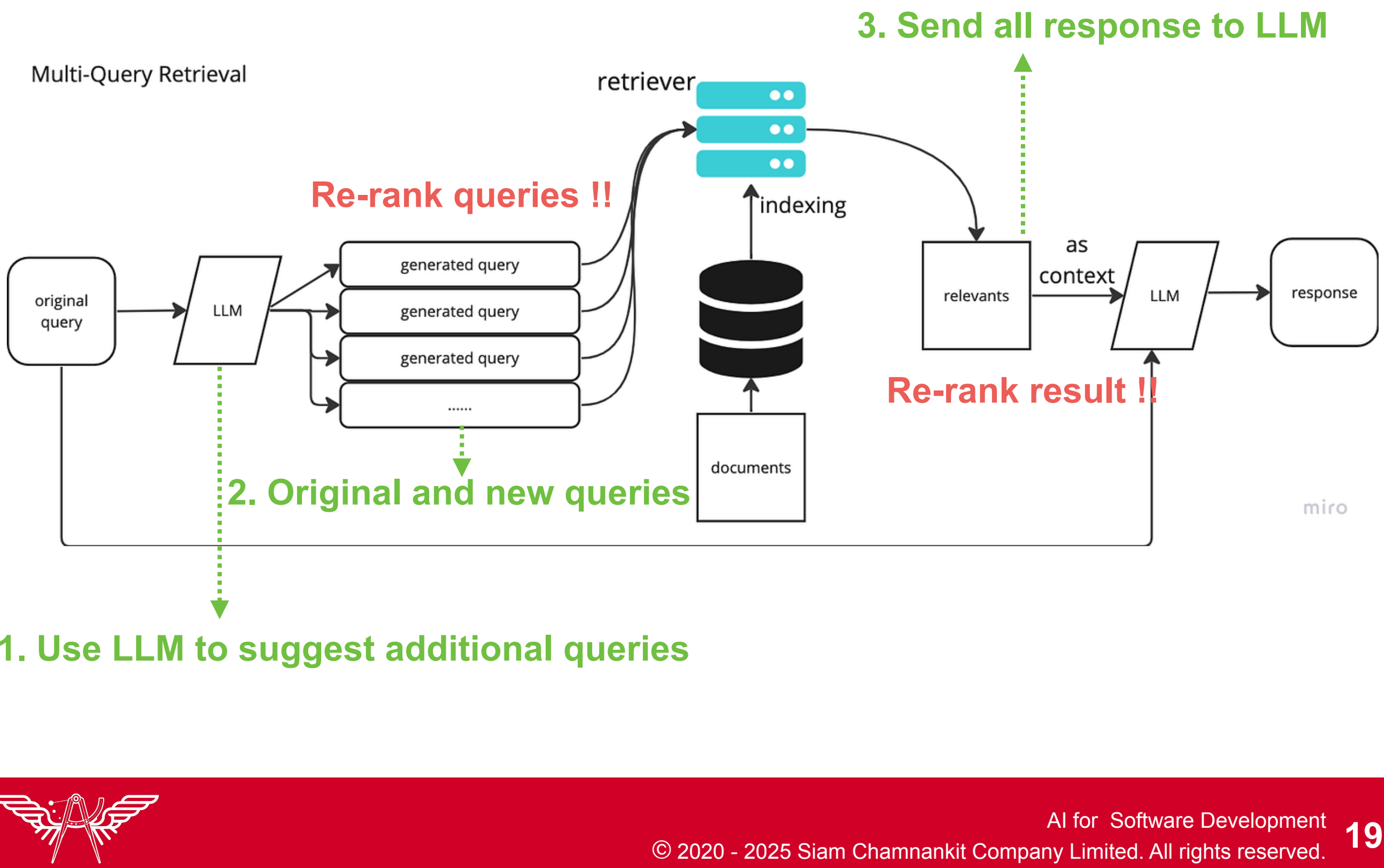
https://arxiv.org/abs/2305.03653

Query Expansion



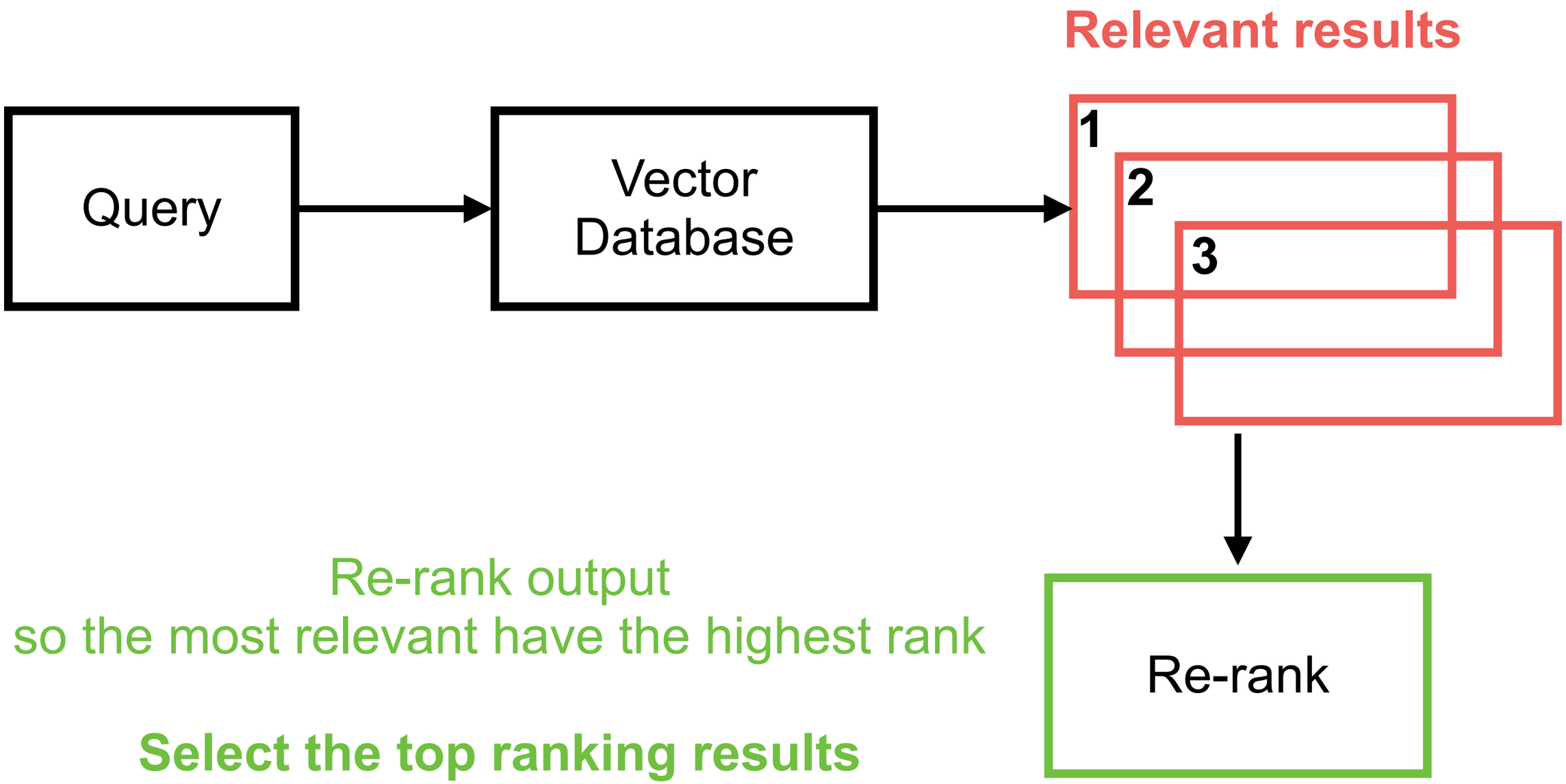
1. Use LLM to suggest additional queries

Re-rank !!



Cross-encoder Re-ranking

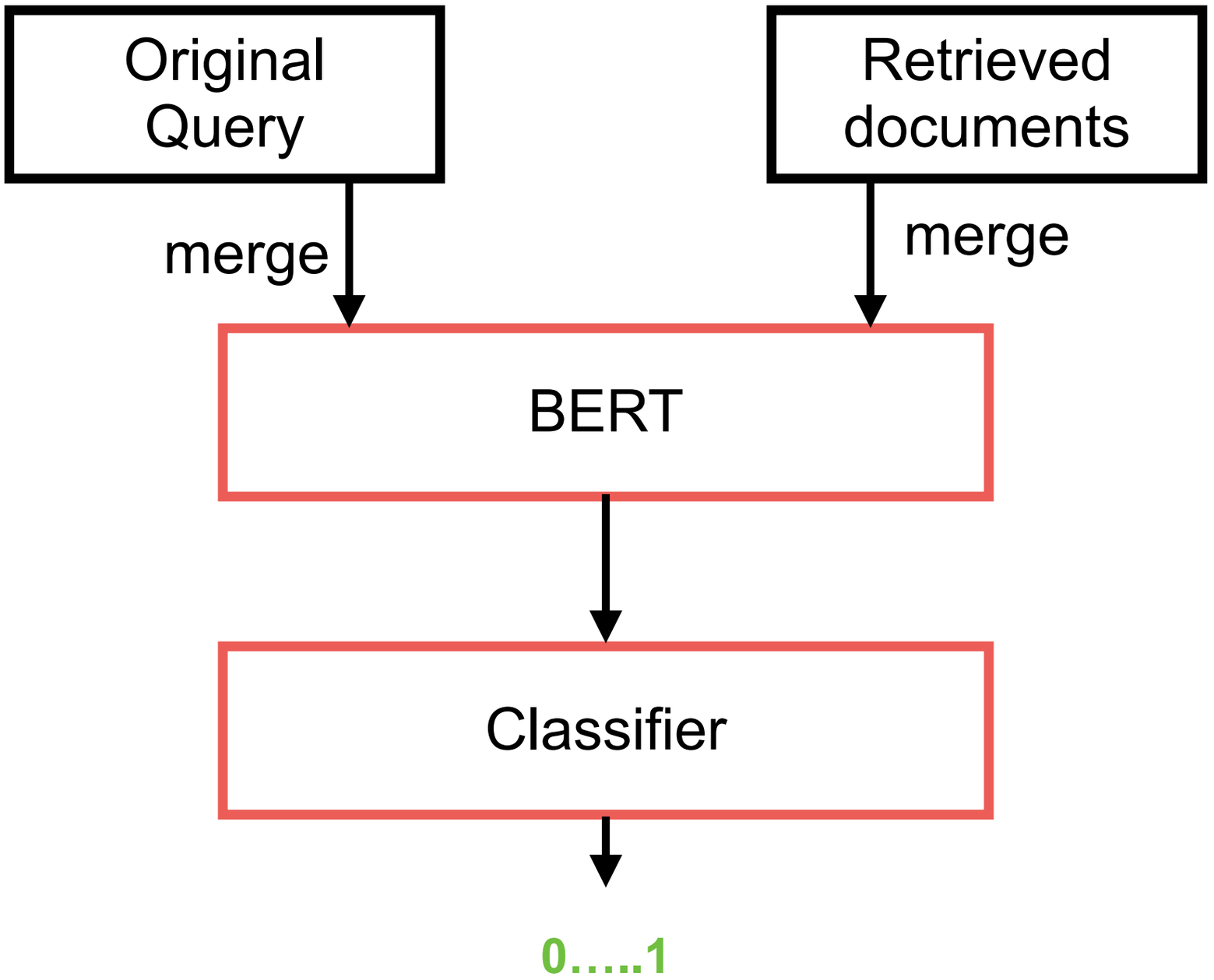
How to ordering relevant results !!



Re-rank output
so the most relevant have the highest rank

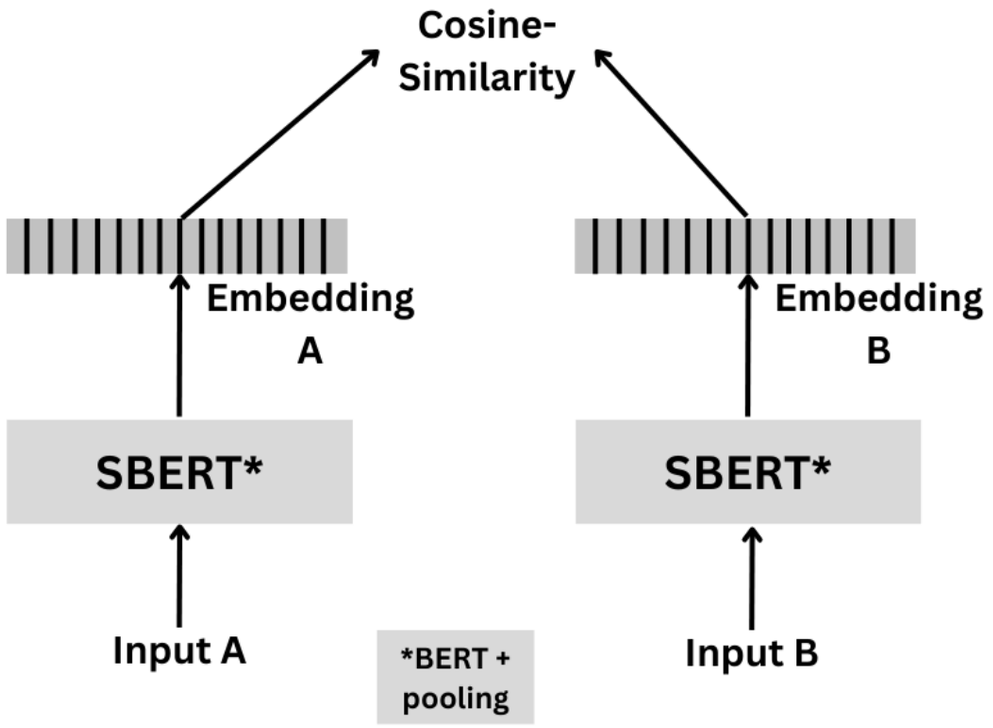
Select the top ranking results

Cross-Encoder

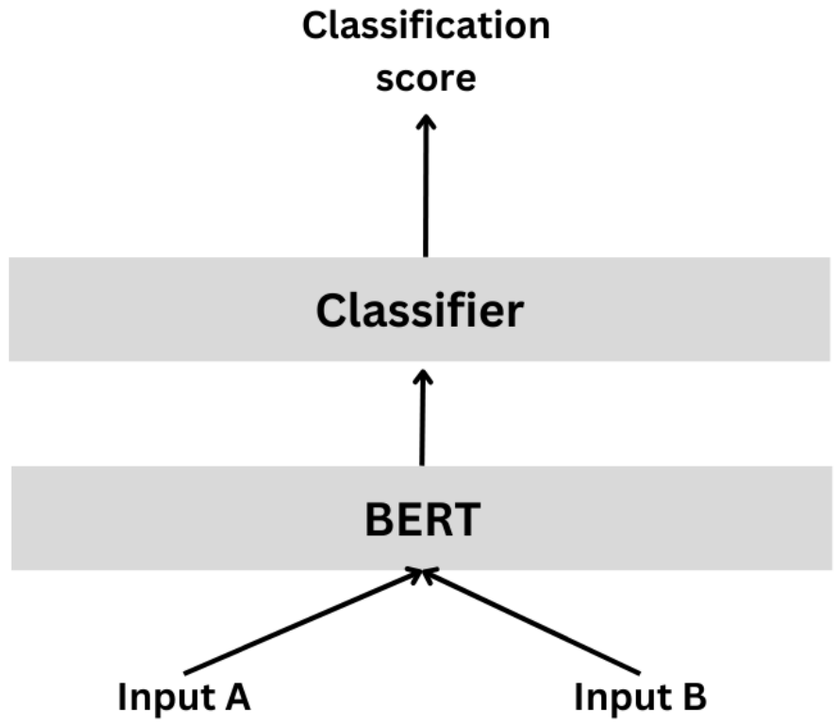


Cross-Encoder !!

Bi-encoder

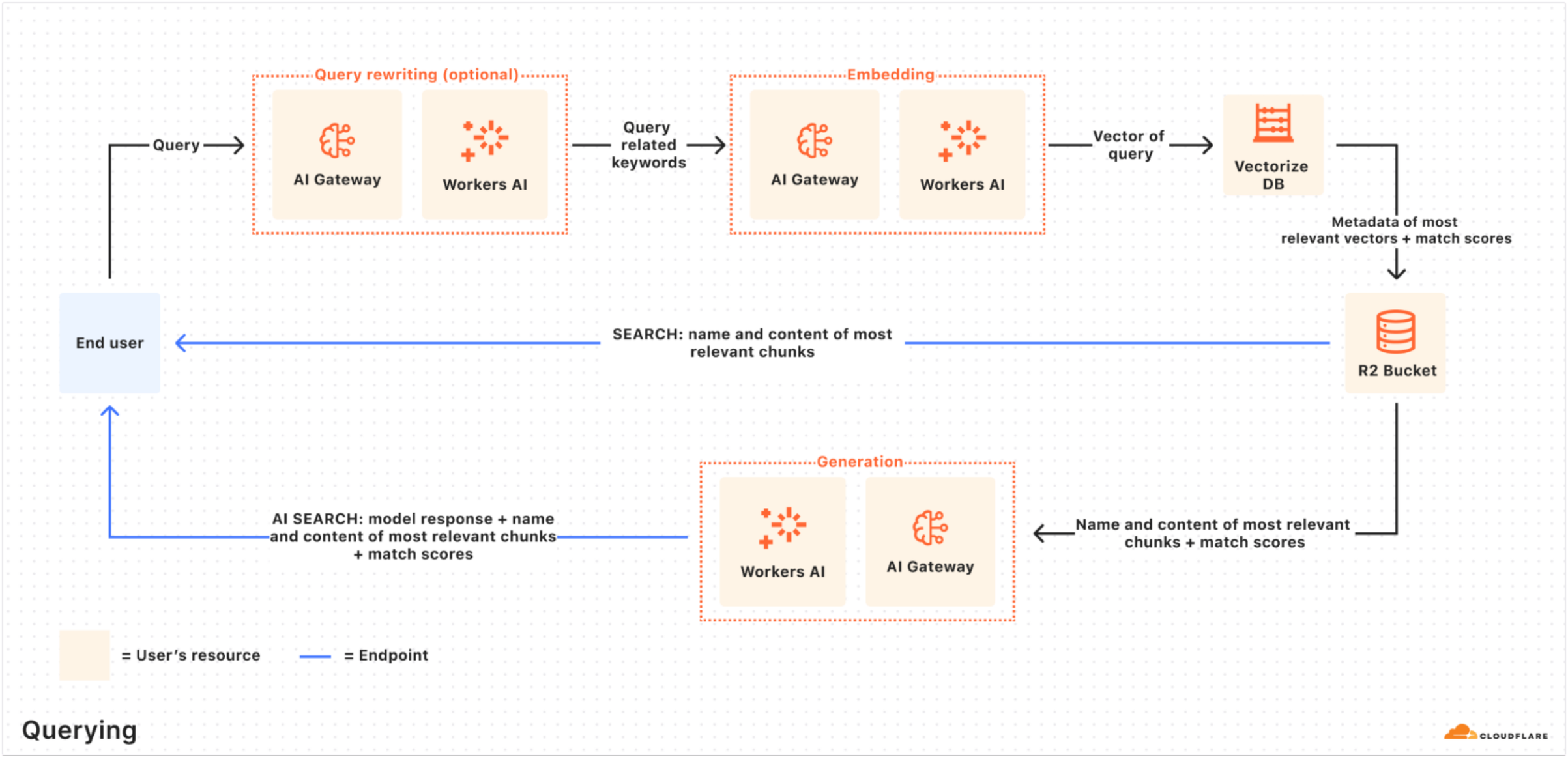


Cross Encoder



https://sbert.net/examples/cross_encoder/applications/README.html

Cloudflare AutoRAG



https://developers.cloudflare.com/autorag/

Guardrails

https://github.com/guardrails-ai/guardrails

Guardrails

Help to build reliable AI applications

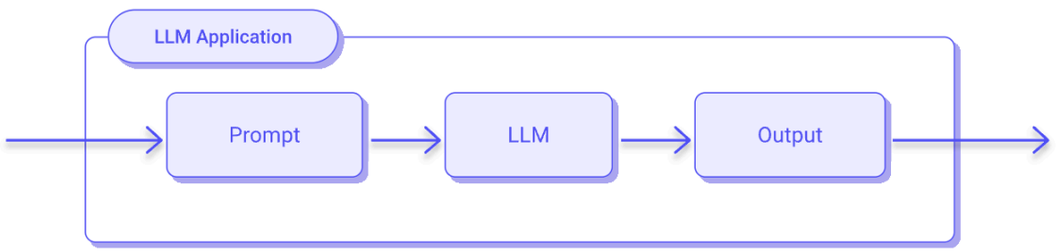
Guard for
input and output

Help to generate
Structured output

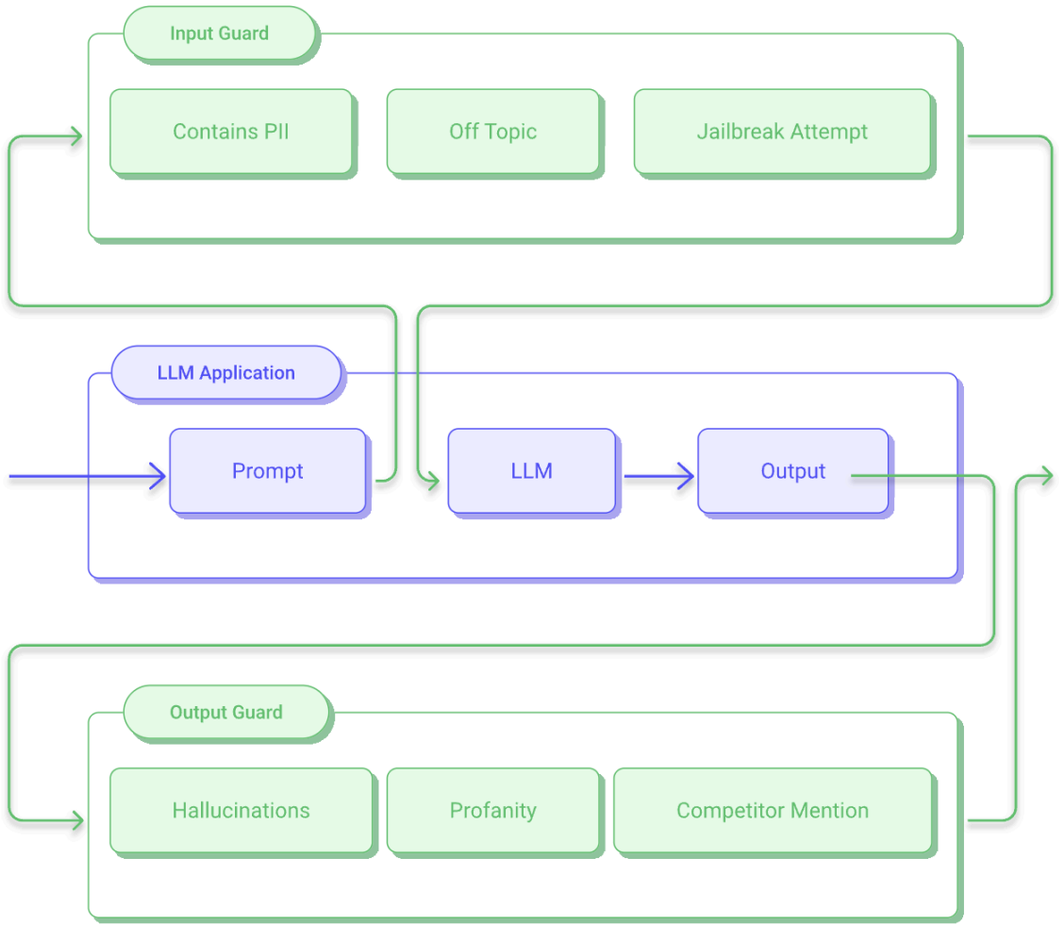
https://github.com/guardrails-ai/guardrails

Guardrails

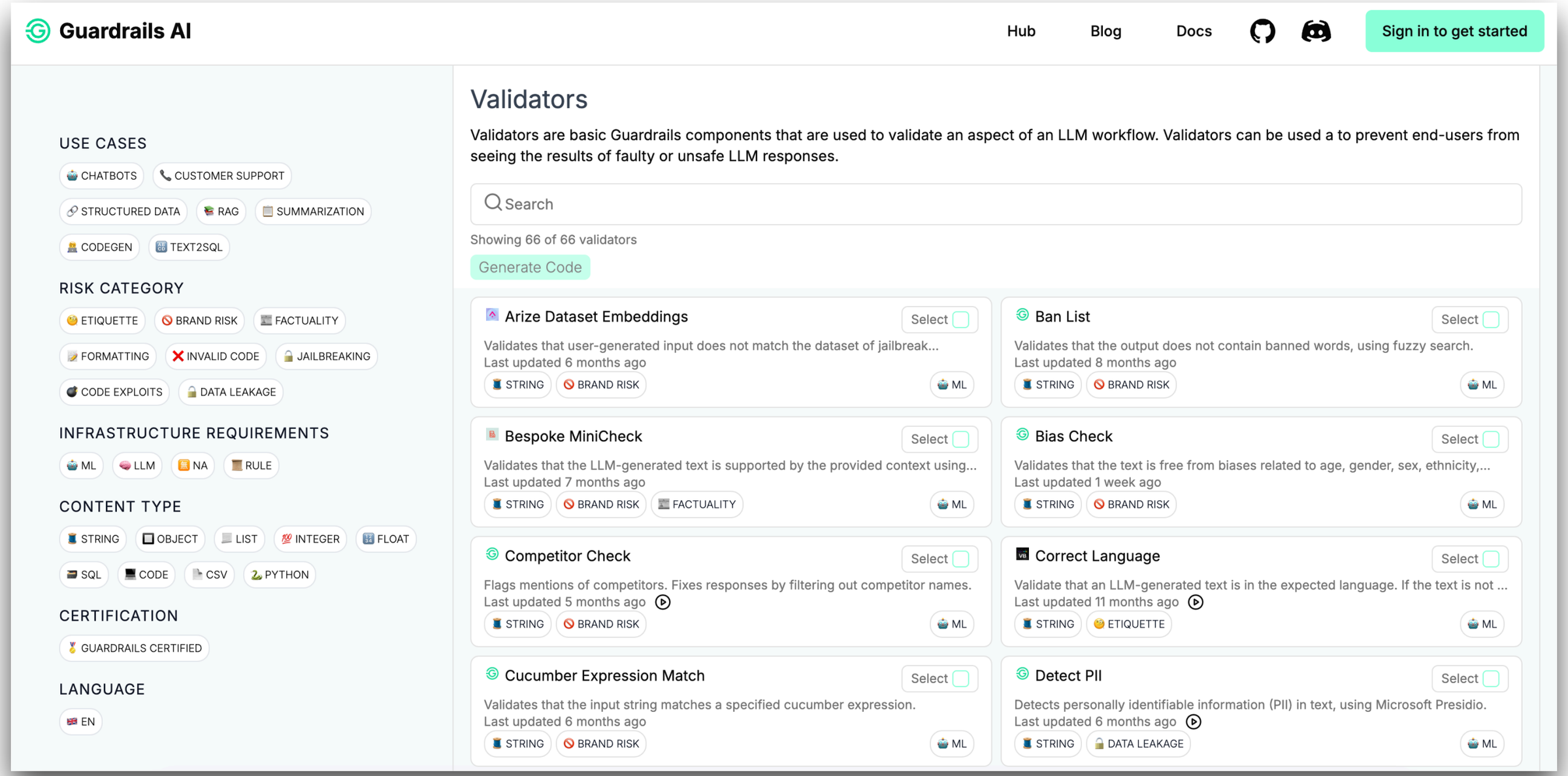
Without Guardrails



With Guardrails

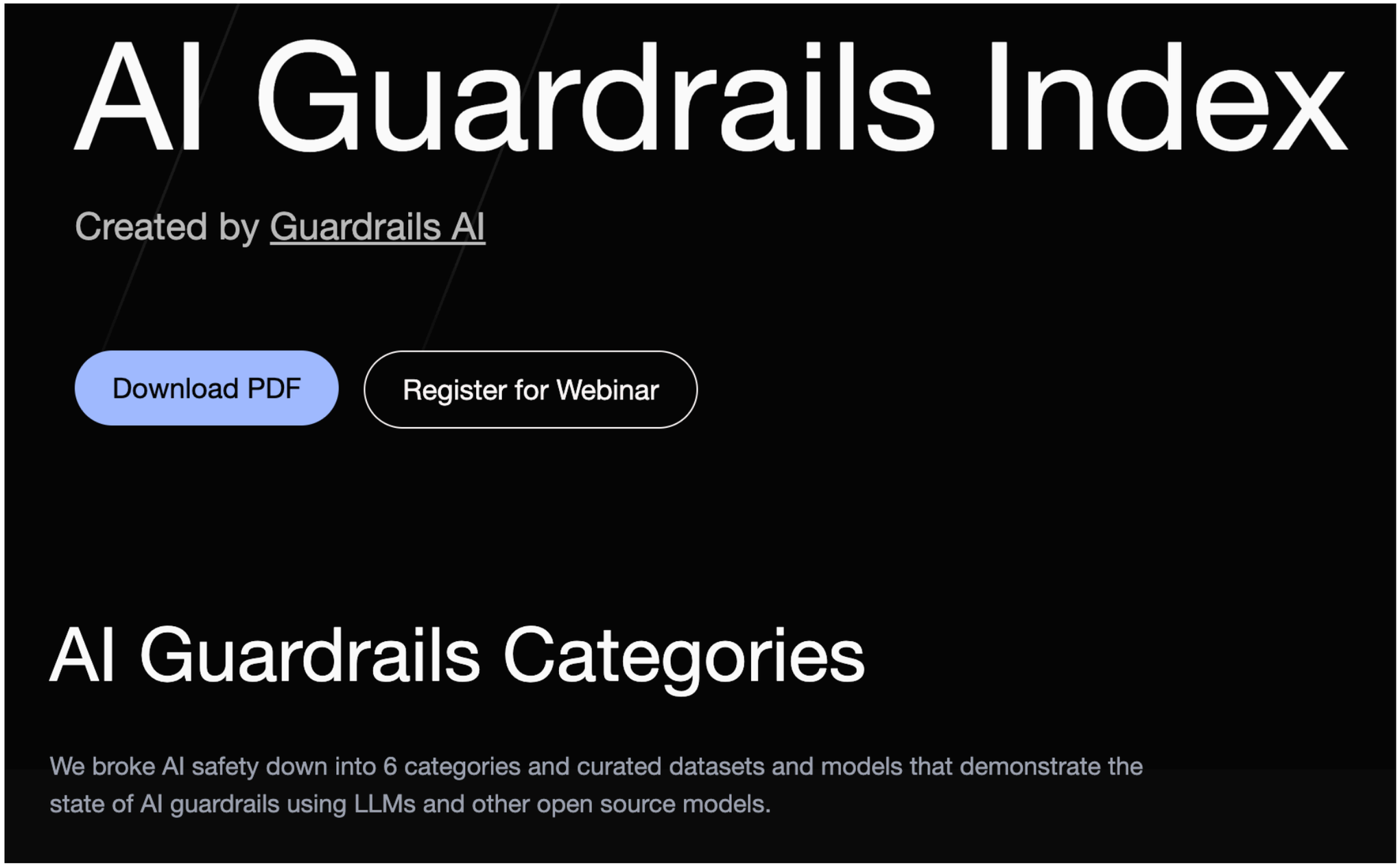


Guardrails Hub



https://hub.guardrailsai.com/

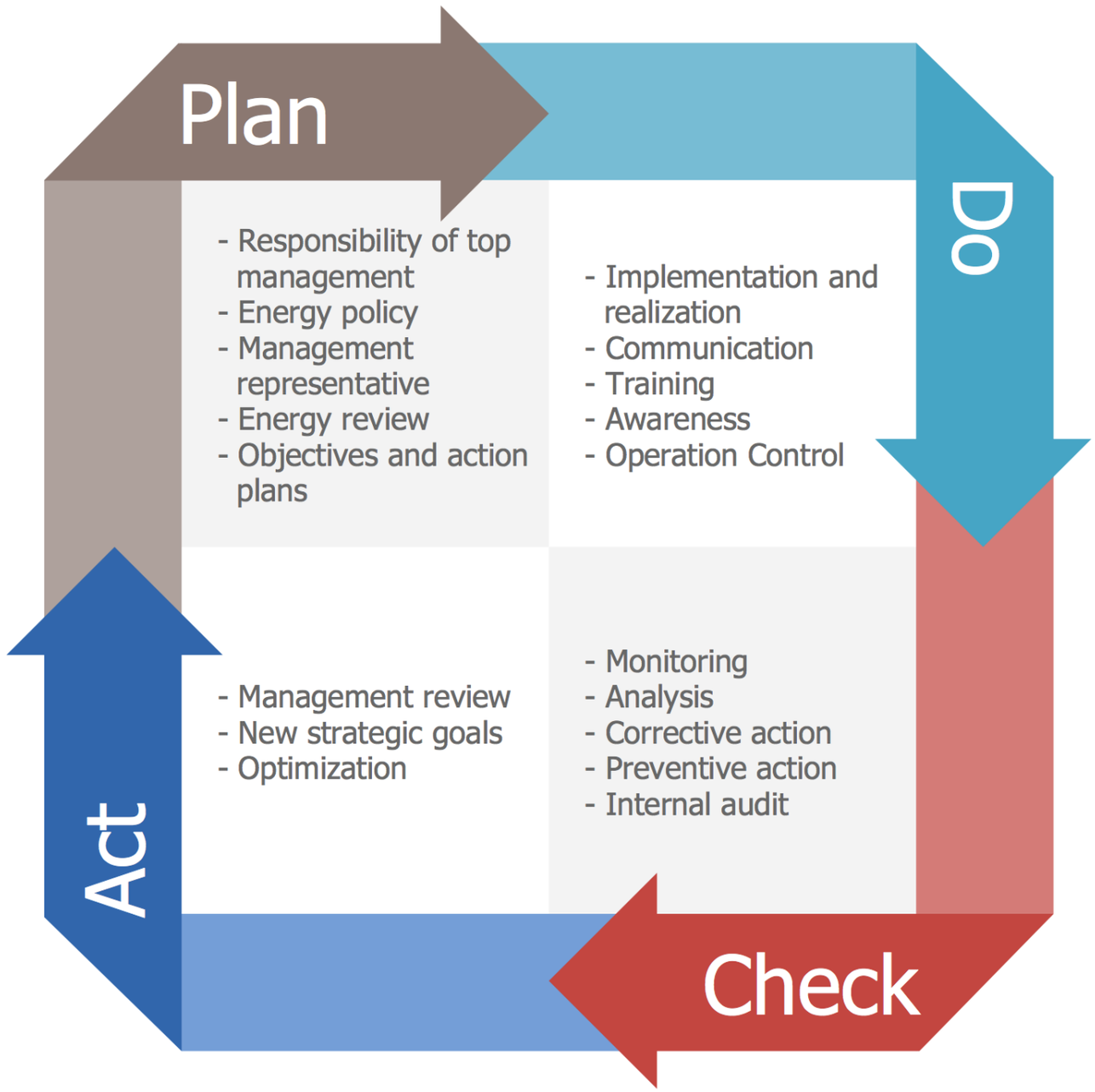
Guardrails Index



https://index.guardrailsai.com/

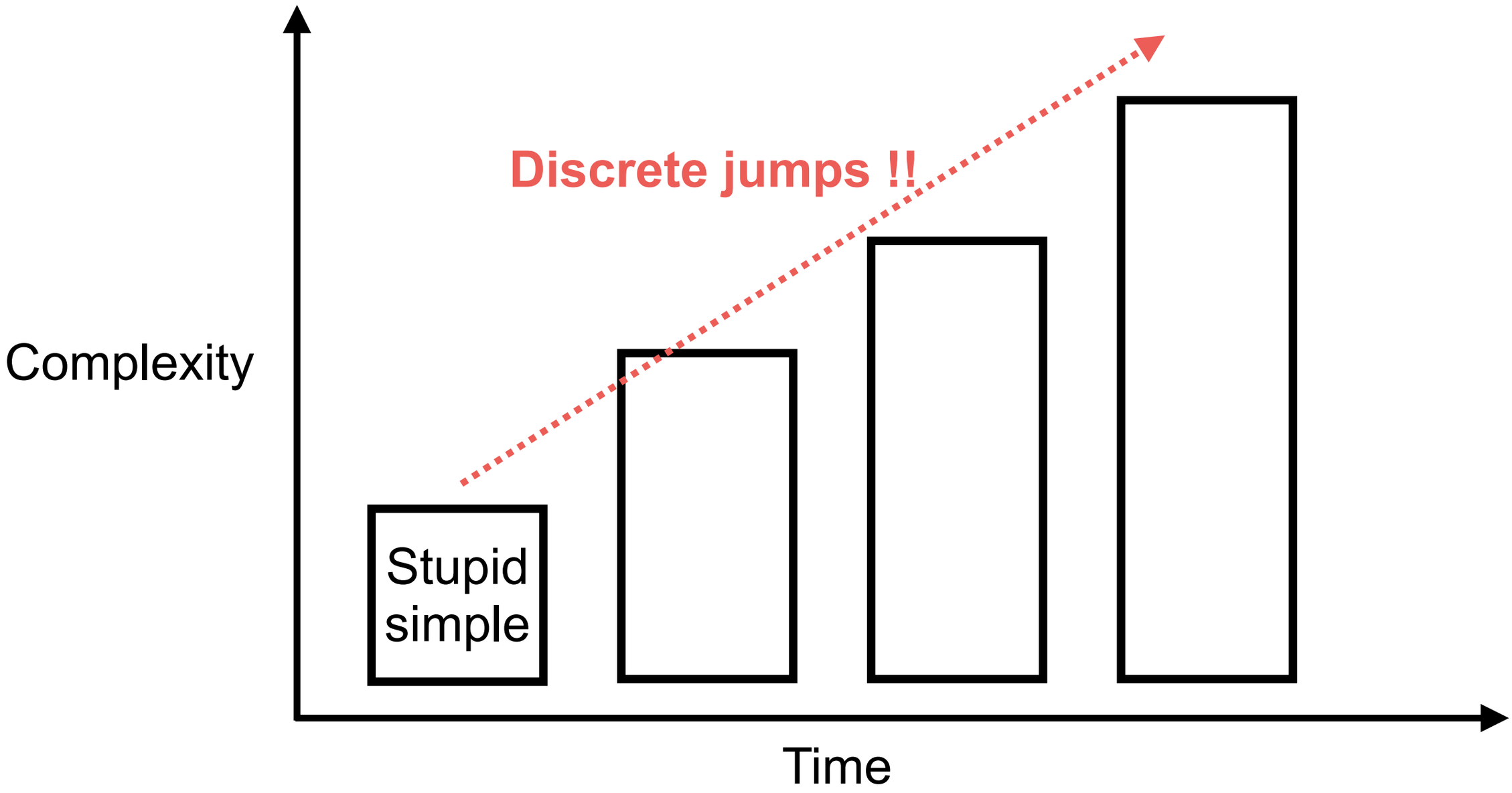
Workshop with IDE

PDCA process

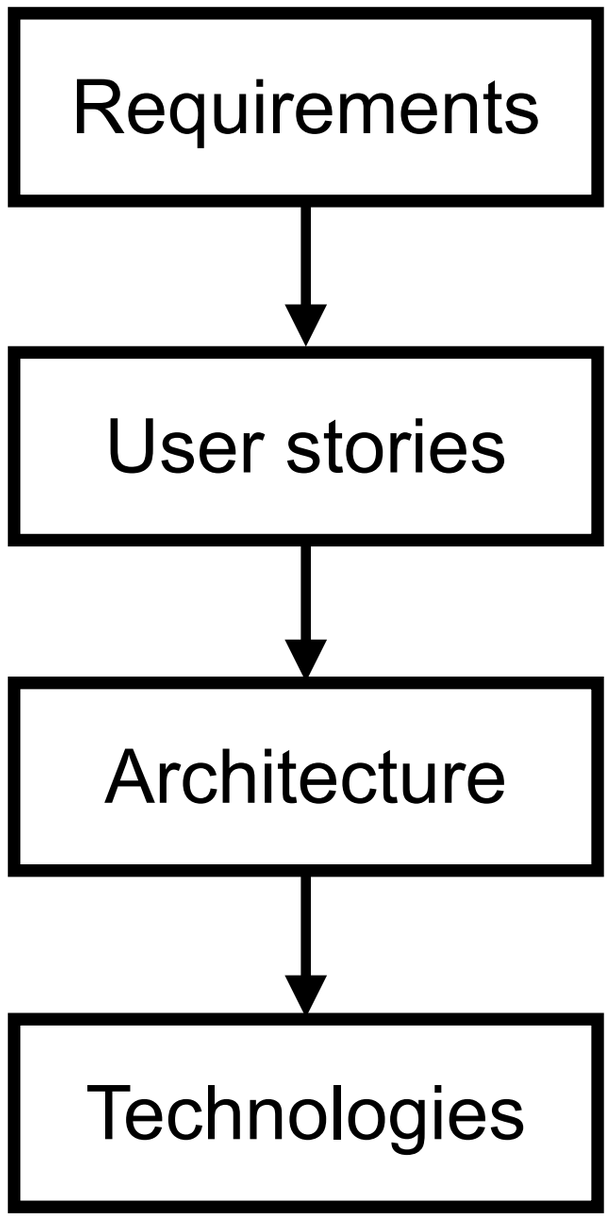


Staged Complexity

Start small and get a feature working first



Planning workflow



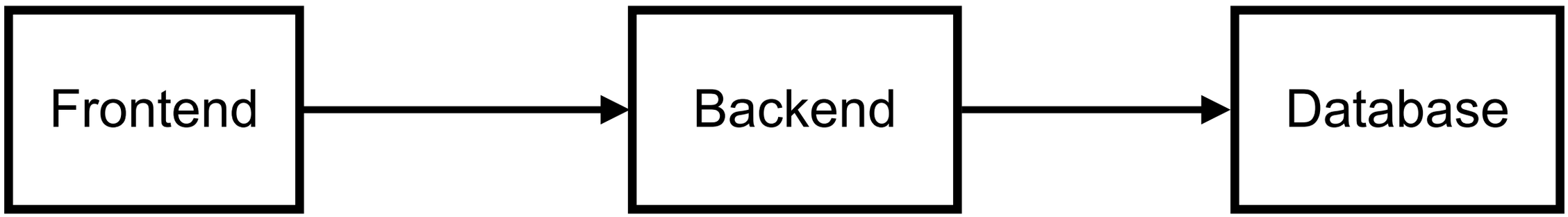
What you’re build ?

What use need ?

Selection ?

Selection ?

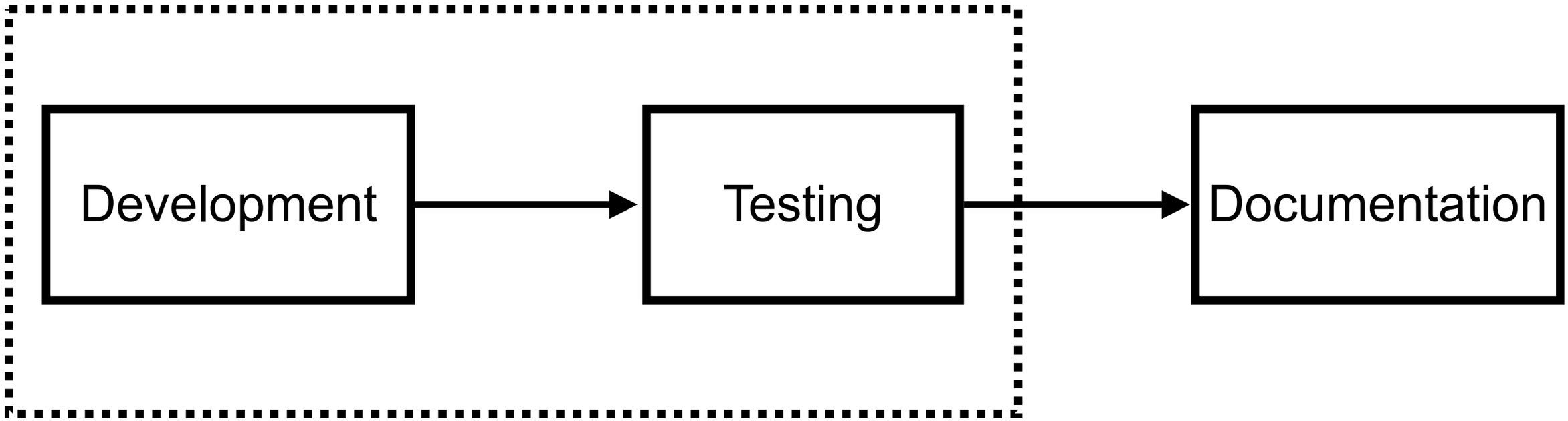
Architecture of project



Choose your tech stack !!

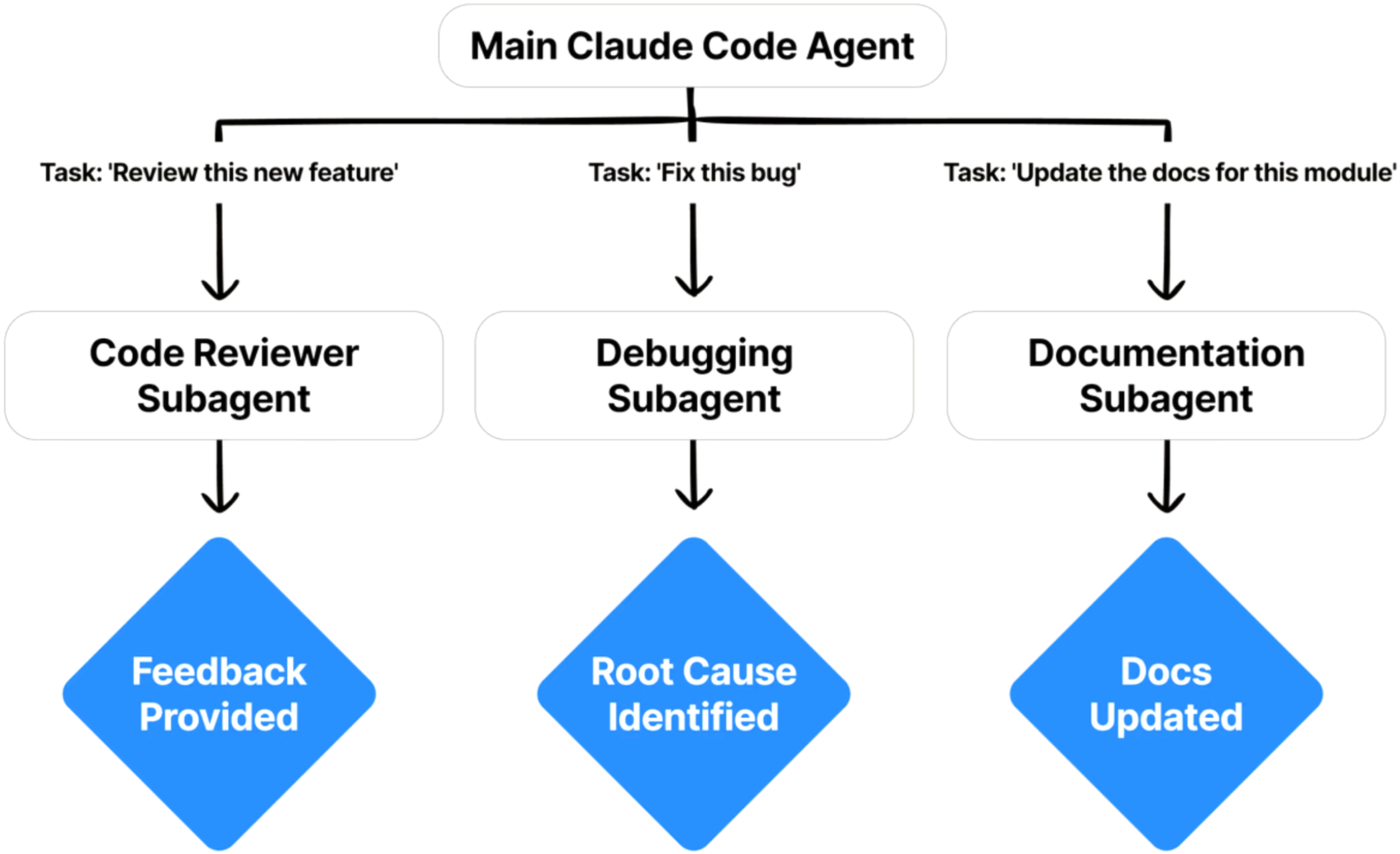
Processes

Iterative and incremental process



AI Subagent

AI Subagent



https://code.claude.com/docs/en/sub-agents

AI Subagent

https://github.com/VoltAgent/awesome-claude-code-subagents

Start your journey