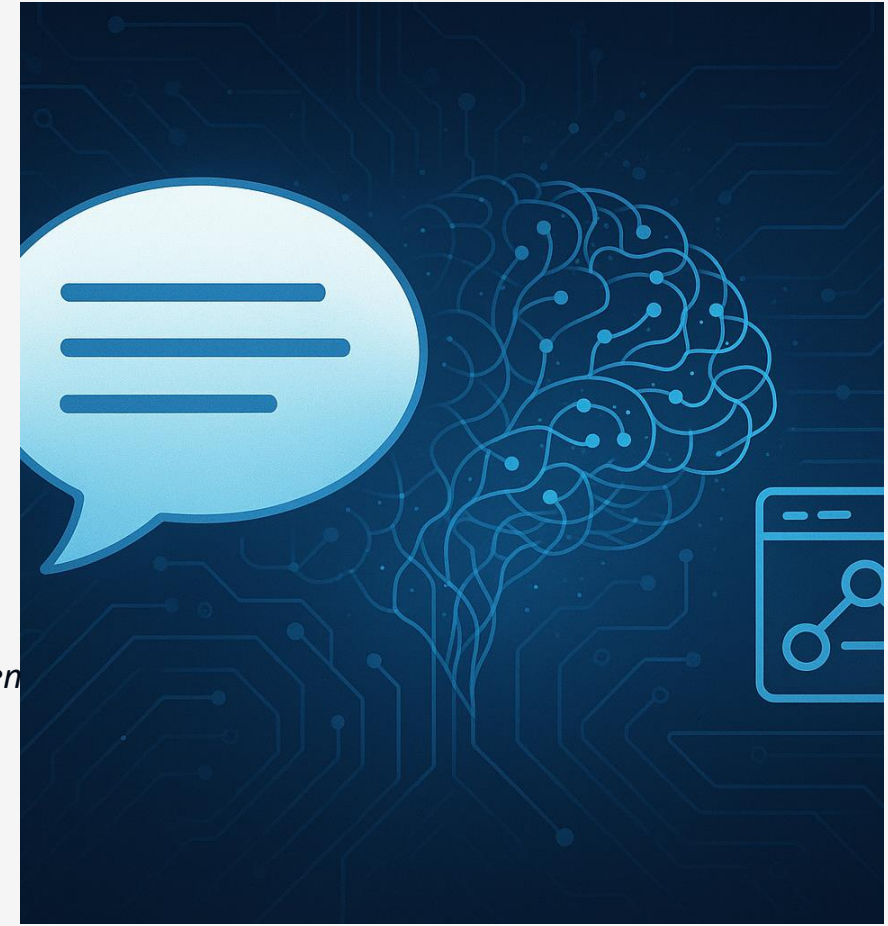
Prompt Engineering for Product Features

Master prompt engineering to create engaging, AI-driven product features.
Learn to craft effective prompts for LLMs and integrate them into apps.

July 30, 2025



Few-Shot Prompting

Guide LLMs with a handful of examples to improve performance on complex tasks.

Zero-Shot

No examples; rely entirely on instructions.

One-Shot

Single example to set format and tone.

Few-Shot

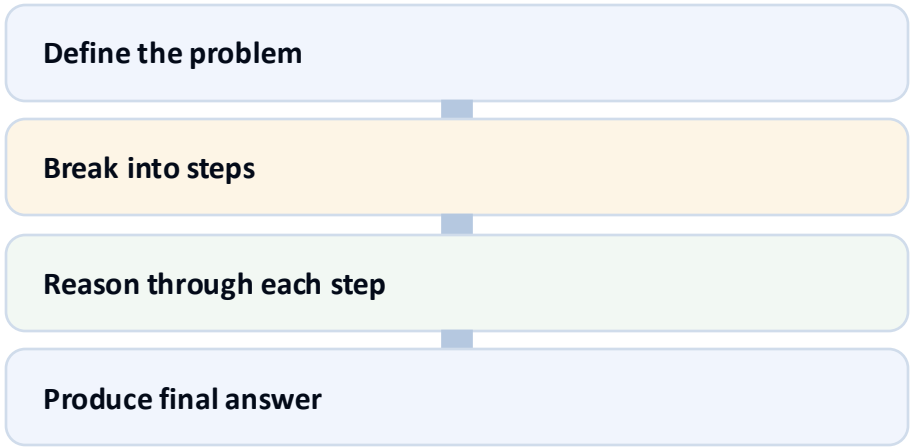
Provide multiple examples to convey patterns.

- In-context learning uses examples (shots) to help models recognise patterns and perform tasks.
- Zero-shot provides no examples; one-shot uses one; few-shot uses multiple examples to improve accuracy.
- Choose the number of shots based on task complexity and format requirements.

[1] [2]

Chain-of-Thought Prompting

Embed reasoning steps in your prompt to guide models through complex problems.



- Adds intermediate reasoning steps to prompts, enabling models to logically work through tasks.
- Decomposes complex problems into manageable parts and guides the model to think step-by-step.
- Improves performance on maths, commonsense reasoning and decision-making tasks by up to 19–24 points in benchmark studies.

[3] [4] [5] [6]

LangChain

Build end-to-end LLM applications with modular chains and tools.

Chains

Compose sequences of LLM calls and tools to accomplish tasks.

Prompt Management

Reusable templates and variables standardise your prompts.

Agents

Autonomously decide actions and call APIs or tools as needed.

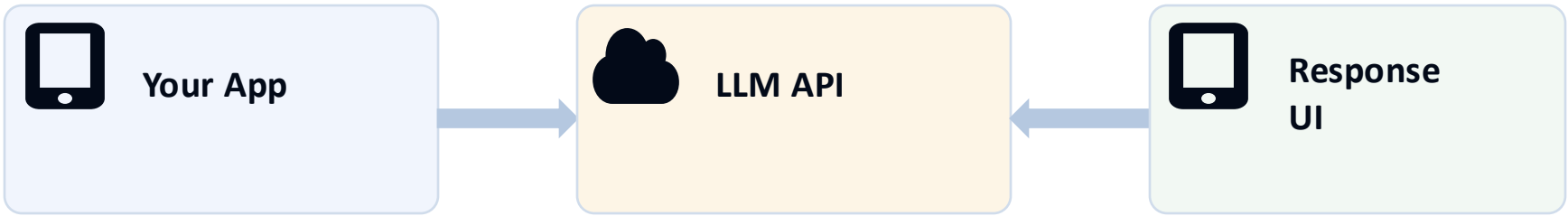
Vector Stores

Store embeddings for similarity search and maintain conversational memory.

[7] [8] [9]

Integrating LLM APIs

Bring AI into your app while balancing latency, cost and privacy.

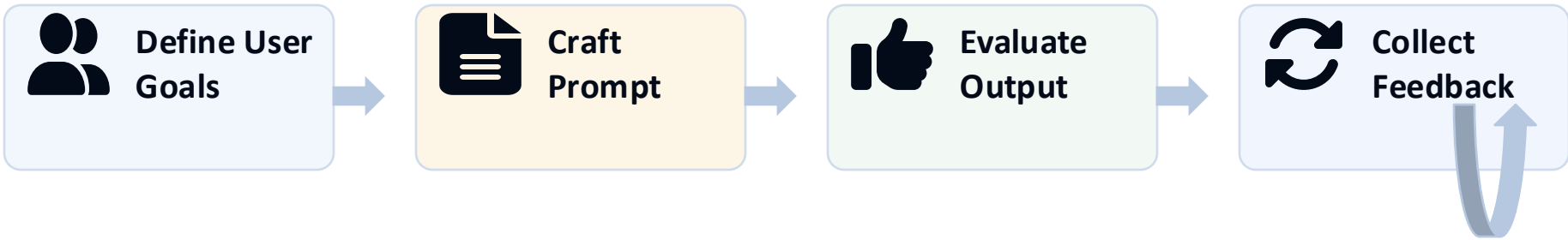


- Set up an API account and securely store keys in environment variables.
- Call the API from your backend, parse responses and deliver them to the UI.
- Monitor token usage and choose appropriate model sizes to balance cost and latency.
- Consider caching and rate limits to reduce latency and manage costs.

[10] [11] [12]

User Feedback & Prompt Iteration

Improve prompts by incorporating user needs and iterative feedback.



- Craft prompts with clarity, context and constraints to improve relevance and consistency.
- Understand user needs through interviews, surveys and usage analytics.
- Track metrics like response relevance, completion rate, time and user satisfaction.
- Iteratively refine prompts based on qualitative feedback and quantitative metrics.

[13] [14] [15] [16] [17]

User-Centric AI Features

Design intelligent features that put user needs first.

Simplicity & Usability

Hide AI complexity and offer intuitive interfaces; avoid adding features without practical value.

Transparency & Control

Give users visibility into how AI works and let them correct or opt out of suggestions.

Personalisation & Predictive

Leverage user data to tailor experiences while respecting privacy and ethics.

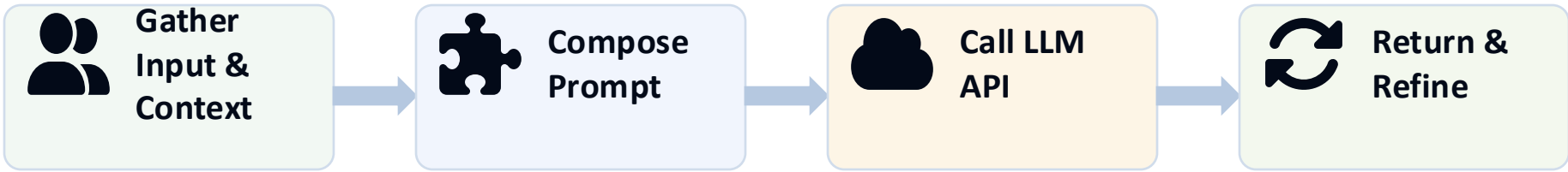
[18] [19] [20] [21] [22]

Error Handling & Trust

Provide fallback options and reporting tools to handle errors and build trust.

Let’s Build!

Now it’s time to apply prompt engineering and LangChain to build a chatbot feature.



- Collect user inputs and relevant context from your product.
- Use LangChain to assemble prompts with few-shot and chain-of-thought techniques.
- Send the prompt to an LLM via API and parse the response for your UI.
- Iteratively refine prompts and features based on user feedback and performance.

[23] [24] [25]