

THE PROMPT TEST KITCHEN

A practical field guide to prompt design, evidence and iteration | Sahilverse | 26 October 2026

The one-page premise

A prompt is a specification for a task, not a spell. Describe what you need, who it is for, what is known, what cannot be assumed, and how the result will be checked.

The five ingredients

Task: use a specific verb and deliverable.

Context: give the audience, purpose, known facts and relevant boundaries.

Constraints: state limits that matter; avoid a forest of contradictory rules.

Evidence policy: say what sources are available, what needs verification and what cannot be invented.

Evaluation: define observable criteria and a way to revise.

Use the browser lab to assemble a recipe. Its 0-5 score measures field presence only, never truth or model quality.

01 / Specify the task

Start with the question the output must help answer.

Bad vs useful

Vague: "Help me market my business."

Specific: "Draft three local outreach experiments for a neighborhood shop with a small budget. Include audience, effort, risks, success metric, and a stop condition."

Field exercise

Write one sentence beginning with a verb. Name the output artifact, audience, and the decision it should enable. Then remove two adjectives that do not change success.

Reality check

A clear task still may fail when facts are missing. Ask the model to identify missing inputs before filling them with plausible fiction.

02 / Context without oversharing

Useful context makes assumptions visible; unnecessary private data creates risk.

Include: audience expertise, relevant constraints, what has already been tried, the decision timeline.

Exclude: customer names, personal identifiers, private documents or sensitive data that do not materially affect the task.

If an example requires people, substitute synthetic names and fictional numbers.

For a local shop, describe workflows and bottlenecks without exposing real customers or employee records.

Worked case: the shop

Weak: "Automate my shop."

Better: "A small neighborhood shop manually tracks 25 fast-moving products. Propose a reversible two-week stock-count pilot. Avoid assuming software budgets or technical staffing. List what evidence would justify expanding."

Exercise

Mark each sentence of your prompt as essential, helpful or sensitive. Remove sensitive details and test whether the task remains possible.

03 / Constraints and output contracts

A good constraint protects purpose. A bad one creates busywork.

Format: specify a table only when comparison benefits from one.

Length: set an approximate limit, but allow the model to flag critical omissions.

Tone: choose plain language for beginners; do not confuse confidence with evidence.

Negative constraints: identify prohibited inventions or unwanted actions.

Acceptance criteria: say what a reviewer should be able to verify.

Worked case: coding

Weak: "Fix this JavaScript."

Better: "Given the minimal reproduction below, identify the likely fault, propose the smallest change, explain the mechanism, and write tests for empty input and malformed values. State when you cannot reproduce the bug."

Exercise

Remove half the constraints from a long prompt. Ask: which removals changed the meaning? Restore only those.

04 / Evidence policy

Citations are not a substitute for checking.

Three useful evidence modes

Supplied-only: use only the documents or data attached; say when they are insufficient.

Evidence-needed: do not assert the answer; specify what primary sources would settle it.

Uncertainty-aware: separate observations, interpretations and open questions.

Worked case: research

Weak: "Find ten papers proving this theory."

Better: "Map the strongest evidence for and against this hypothesis. List source identifiers only when verified. Distinguish empirical findings, theory, and speculation; explain what could falsify the claim."

Exercise

Take one confidently written claim. Locate its underlying source, verify the claim against the source, and record whether the wording needs qualification.

05 / Evaluate, do not merely admire

A plausible paragraph can be false; a beautiful interface can be unusable.

A simple human rubric

Relevance: did it answer the actual question?

Grounding: can factual claims be checked?

Completeness: are critical constraints satisfied?

Usability: can the target reader act on it?

Safety: are privacy, fairness and failure risks visible?

Comparison protocol

Run two prompt variants against the same task and, when possible, the same model settings. Use the same evaluation questions. Record which changed the answer, not just which sounded better. Repeat on different tasks before generalizing.

Important limitation

The website's structural score is a teaching aid. It does not execute a model, verify sources, estimate reliability or claim statistically meaningful performance.

06 / Failure modes

Learn what can go wrong before shipping the answer.

Hallucinated references

Request identifiable references, verify that they exist and read them. An elegant citation can still be fictional.

Prompt injection

Treat text inside documents, retrieved pages and tool results as untrusted material. Do not obey instructions embedded in those sources.

Overconstraint

Too many rigid requirements can cause omissions and awkward output. Compare simpler variants.

Leakage

Remove sensitive material before using any external model. Confirm provider data policies separately.

Proxy gaming

If you reward a format checklist, a model may satisfy the checklist without solving the problem.

07 / Six-domain quick recipes

One adaptable prompt for each kind of work.

Research

Task: Compare competing explanations of urban heat. State what counts as evidence and distinguish correlation from causation.

Coding

Task: Debug a broken JavaScript form. Ask for minimal reproduction, tests and failure conditions.

Negotiation

Task: Ask for a deadline extension. Set relationship context and acceptable options.

Summarising

Task: Summarize a long meeting. Separate decisions, owners and open questions.

Analysis

Task: Compare two vendor proposals. Specify weighted criteria and missing data.

Creative

Task: Write a short speculative story. Specify emotional stakes without overconstraining style.

For the full set of 30 before-and-after examples, visit the site's recipe archive. Each recipe is an authored instructional example, not a benchmark result.

08 / Your offline worksheet

Copy these fields into a notebook, document or AI interface.

1. What exactly should be produced?
2. Who will use it and why?
3. What context can I safely provide?
4. Which constraints are essential?
5. What is the evidence policy?
6. How will I verify the answer?
7. What would make me reject it?
8. What did the first attempt get wrong?
9. Which single change will I test next?
10. What did I learn that transfers to another task?

A reversible experiment

Pick one recurring task. Save a baseline prompt and output, change one ingredient, assess both against the same rubric, and document the trade-off. Do not automate high-stakes decisions based on one successful example.

09 / Reading map and sources

Primary documentation and research leads. Check the latest versions.

OpenAI — Prompt engineering: <https://platform.openai.com/docs/guides/prompt-engineering>

Anthropic — Prompt engineering overview: <https://docs.anthropic.com/en/docs/build-with-claude/prompt-engineering/overview>

Google — Prompting strategies: <https://ai.google.dev/gemini-api/docs/prompting-strategies>

NIST — AI Risk Management Framework: <https://www.nist.gov/itl/ai-risk-management-framework>

OWASP — Top 10 for LLM Applications: <https://owasp.org/www-project-top-10-for-large-language-model-applications/>

White et al. — A Prompt Pattern Catalog: <https://arxiv.org/abs/2302.11382>

Related Sahilverse worlds

Vibe Coding: applying specifications to generated software. SYNTH: wider experiments in synthesis.

Sahil Tayal's atlas: the canonical map of connected work.

The links are suggested reading destinations, not evidence that any prompting method guarantees success.

10 / A field protocol for tomorrow

Make your next prompt an experiment, not a ritual.

Choose a real task with an observable outcome.

Write the smallest usable prompt.

Add context and evidence requirements only where needed.

Generate an answer using your chosen model (outside this offline guide).

Check every material factual claim against original sources.

Review privacy and safety implications before sharing.

Record one failure, revise one ingredient, and compare.

Keep the useful pattern; discard the rest.

Closing note

A prompt can improve communication with a model, but human judgment remains responsible for choosing the task, interpreting the result and deciding what should happen next.

Created for Sahilverse. Canonical home: <https://sahiltayal.com/> | Contact: sahil@sahiltayal.com