# GPT‑5 Router‑Optimized Universal Prompt Pack (v1.1)

**What this is:** A field‑tested, router‑aware prompt pack tuned for GPT‑5.

**How to use:** Paste the **Router Boost Header 2.0** above any task below, then use the upgraded prompt. Each item includes a fast audit (strengths, gaps, tuning) so you know \*why\* it works.

――――――――――――――――――――――――

## Router Boost Header 2.0 (paste above any prompt)

**Task**: [one sentence describing “done”].

**Context/Grounding**: [paste facts/links/notes]. Cite sources if summarizing; don’t invent.

**Constraints**: audience=[…], tone=[…], length=[…], locality=[region/laws], non‑negotiables=[…].

**Output Contract**: [exact format/schema; if JSON, include a schema].

**Tool Grants**: You may use internal reasoning, code execution, and structured output. **Do not** expose chain‑of‑thought; return only the final results.

**Mode**: Choose `fast` for simple tasks, `deep` for complex ones; state the choice on one line before the output.

**Self‑Check**: Validate constraints, factuality (vs. sources), and format before returning. If JSON, ensure it parses.

**Failure Policy**: If blocked or context is thin, list missing info and ask **3 sharp questions**; otherwise proceed with explicit assumptions labeled “Assumptions.”

> Tip: Keep the header short in production—only include fields that matter. If you need determinism, ask for “low‑randomness; no lateral riffs.”

――――――――――――――――――――――――

# Universal GPT‑5 Prompt Pack v1.1

Below: for each prompt

* **Use when**: best fit.
* **Strengths**: what’s good already.
* **Gaps**: what to tighten for GPT‑5.
* **Router tuning**: small switches that improve results.
* **Upgraded prompt**: copy/paste ready.
* **(Optional) Strict JSON variant**: when you need machine‑readable output.

――――――――――――――――――――――――

## 1) Executive Summary (Any Topic)

**Use when:** You need crisp, executive‑level clarity in 30–90 seconds.

**Strengths:** Forces prioritization; covers timing and action.

**Gaps:** Can drift into fluff; doesn’t enforce one‑line bullets; missing “evidence”.

**Router tuning:** Demand one‑line bullets with bold labels; add “evidence” blip; enforce count.

**Upgraded prompt**

Create exactly \*\*5 one‑line bullets\*\* summarizing [topic/brief].  
Each bullet starts with a bold label: \*\*What matters\*\*, \*\*Why now\*\*, \*\*Risks\*\*, \*\*Decision\*\*, \*\*Next actions\*\*.  
Add ≤12 words per bullet. Include 1 source or metric if available.  
Mode: [fast/deep]. Return as a simple bullet list—no preamble.

**Strict JSON variant**

Return valid JSON:  
{ "what\_matters": "...", "why\_now": "...", "risks": "...", "decision": "...", "next\_actions": "..." }

――――――――――――――――――――――――

## 2) Research Plan (Adversarial)

**Use when:** You must test a claim/feature beyond happy‑path.

**Strengths:** Calls for metrics, data, adversarial tests.

**Gaps:** No threat model; no instrument plan; no stop/continue math.

**Router tuning:** Introduce threat model + falsification criteria; add power checks.

**Upgraded prompt**

Design an \*\*adversarial research plan\*\* to evaluate [claim/feature]. Include:  
1) Objectives & hypotheses (null + alt); 2) Success metrics & thresholds; 3) Threat model (abuse, edge cases);  
4) Data to collect (fields, sample size/power);  
5) Protocols (A/B, holdout, offline evals);  
6) Adversarial tests & red‑team scripts;  
7) Stop/continue rule with math;  
8) Reporting template (tables/plots).  
Mode: [fast/deep]. Output as a numbered outline.

――――――――――――――――――――――――

## 3) Decision Memo

**Use when:** A one‑pager to choose among options.

**Strengths:** Options, costs, risks, reversibility, rec.

**Gaps:** No owner/date format; no “evidence” box; weak contingency.

**Router tuning:** Add RACI owner/date; add 30/60/90 follow‑ups.

**Upgraded prompt**

Write a one‑page decision memo for [choice]. Include:  
- Context (1 para) with constraints & evidence;  
- Options (3): summary, costs (one‑time/run), risks, reversibility;  
- Recommendation: \*\*one\*\* choice with rationale;  
- Owner + Decision date; 30/60/90‑day checkpoints;  
- Contingency triggers & rollback plan.  
Mode: [fast/deep]. Keep ≤400 words.

――――――――――――――――――――――――

## 4) Project Plan One‑Pager

**Use when:** Turn messy notes into plan.

**Strengths:** Scope, milestones, owners, risks, comms, RAID.

**Gaps:** No critical path; RAID often hand‑wavy.

**Router tuning:** Add dates & simple Gantt list; RAID as compact table.

**Upgraded prompt**

From these notes: [paste], produce a one‑page plan with:  
1) Scope (in/out);  
2) Milestones (name, owner, date) in order;  
3) Critical path (1‑3 bullets);  
4) Comms cadence (who, channel, freq);  
5) RAID summary table (Risk/Assumption/Issue/Dependency → owner, impact, mitigation);  
6) Acceptance criteria (bullet list).  
Mode: [fast/deep]. Keep it skimmable.

――――――――――――――――――――――――

## 5) Meeting → Decisions

**Use when:** Converting raw notes to what matters.

**Strengths:** Decisions & actions separation.

**Gaps:** No owners on decisions; action status taxonomy missing.

**Router tuning:** Add decision owner + rationale; status enum.

**Upgraded prompt**

Convert these notes: [paste] into:  
A) \*\*Decisions\*\* list (decision, owner, rationale, date);  
B) \*\*Actions\*\* table {owner, step, due, status ∈ [New, In‑Progress, Blocked, Done]}.  
Mode: [fast/deep]. No commentary, just the two sections.

**Strict JSON variant**

{ "decisions": [ { "decision": "", "owner": "", "rationale": "", "date": "" } ],  
 "actions": [ { "owner": "", "step": "", "due": "", "status": "New|In-Progress|Blocked|Done" } ] }

――――――――――――――――――――――――

## 6) Cold Email Trio

**Use when:** 3‑touch outbound sequence.

**Strengths:** Problem → proof → ask. Short.

**Gaps:** ICP nuance; weak personalization; missing CTA micro‑asks.

**Router tuning:** Insert first‑line personal hook; vary asks.

**Upgraded prompt**

Write \*\*3 cold emails\*\* for [offer] to [ICP].  
Email 1: name the \*\*patterned pain\*\*; end with a 10‑min micro‑ask.  
Email 2: social proof/insight (number/metric), 1 sentence case study.  
Email 3: crisp ask with 2 time options.  
Each ≤120 words, 5‑7 sentences, no fluff. Include a {First‑line personalization} placeholder.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 7) LinkedIn Authority Post

**Use when:** Thought leadership for execs + builders.

**Strengths:** Structure, framework, prompt.

**Gaps:** Risk of buzzwords; no proof.

**Router tuning:** Require 1 mini‑case and 1 number.

**Upgraded prompt**

Write a LinkedIn post on [topic] for execs + builders:  
- 3 punchy paragraphs (≤60 words each);  
- 1 mini‑framework (3 bullets, named);  
- 1 thought prompt (1 line);  
- Include one concrete number or example; avoid buzzwords.  
Mode: [fast/deep]. No hashtags unless asked.

――――――――――――――――――――――――

## 8) X Post (Bold, No Hashtags)

**Use when:** High‑signal micro‑take.

**Strengths:** Tight character limit, bold stance.

**Gaps:** Might overrun chars; no proof token.

**Router tuning:** Enforce count; include 1 fact word/number.

**Upgraded prompt**

Write one confident X post on [insight/news]. ≤240 chars.  
Format: HOOK — TAKEAWAY. Include \*\*one\*\* concrete fact or number.  
No hashtags. No emoji at the end. Mode: [fast/deep].

――――――――――――――――――――――――

## 9) YouTube Kit

**Use when:** Fast ideation + structure.

**Strengths:** Titles, open, chapters.

**Gaps:** Title length drift; missing viewer promise.

**Router tuning:** Enforce title count/length; add “who it’s for.”

**Upgraded prompt**

For a video on [topic], produce:  
- \*\*10 titles\*\* (<60 chars);  
- A two‑sentence cold open that states who it’s for and the promise;  
- Chapter list with timestamps (estimate) and outcomes per chapter.  
Mode: [fast/deep]. No clickbait lies.

――――――――――――――――――――――――

## 10) Content Angle Generator

**Use when:** Topic expansion without repetition.

**Strengths:** Rich buckets.

**Gaps:** Duplicates; vague angles.

**Router tuning:** Enforce uniqueness + sample headline.

**Upgraded prompt**

List \*\*25 distinct content angles\*\* for [niche/product] across:  
how‑to, contrarian, teardown, story, data, tutorial, tool, myth vs fact.  
For each: 1‑line angle + a sample headline. No repeats. Mode: [fast/deep].

――――――――――――――――――――――――

## 11) Product Spec from Idea

**Use when:** Move from idea to v1.

**Strengths:** Users, JTBD, metrics, scope.

**Gaps:** Test plan vague; acceptance criteria missing.

**Router tuning:** Add measurable acceptance + de‑scoping rules.

**Upgraded prompt**

Turn this idea into a lean product spec:  
- Users & JTBD; key use cases;  
- Success metrics (leading/lagging) with targets;  
- V1 scope (must/should/could) and out‑of‑scope;  
- Acceptance criteria (measurable);  
- Test plan (happy path, edge, abuse).  
Mode: [fast/deep]. ≤500 words.

――――――――――――――――――――――――

## 12) UX Critique

**Use when:** Actionable UI improvements.

**Strengths:** Issues + fixes.

**Gaps:** Evidence often light; microcopy not tested.

**Router tuning:** Severity scale + before/after microcopy.

**Upgraded prompt**

Critique the UX of [flow/screen]. Deliver:  
- 10 issues with severity ∈ {P0, P1, P2}, evidence, and concrete fix;  
- A before→after microcopy table (3–5 rows);  
- One quick win and one deeper redesign note.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 13) CSV Data Brief

**Use when:** Shape an analysis plan before coding.

**Strengths:** Questions → steps → visuals.

**Gaps:** Schema ambiguity; data checks missing.

**Router tuning:** Add sanity checks + exact chart types.

**Upgraded prompt**

Given CSV schema: [columns], produce:  
1) 5 decision‑driven questions;  
2) Validation checks (types, nulls, outliers);  
3) Analysis steps;  
4) Exact visuals/tables to produce (chart type, axes, groupings).  
Mode: [fast/deep]. No code unless asked.

――――――――――――――――――――――――

## 14) Code from Spec

**Use when:** From spec to runnable core.

**Strengths:** Architecture, snippets, tests, edges.

**Gaps:** Env assumptions; complexity unbounded.

**Router tuning:** Pin language/runtime; include complexity notes.

**Upgraded prompt**

Given this spec: [paste], provide:  
- Architecture diagram (text) and key components;  
- Core code snippets in [language/runtime] with minimal deps;  
- Tests (unit/integration) and fixtures;  
- Failure/edge cases + graceful handling;  
- Complexity & trade‑offs section.  
Mode: [fast/deep]. Keep idiomatic.

――――――――――――――――――――――――

## 15) Code Review + Refactor

**Use when:** Improve safety & clarity with a plan.

**Strengths:** Smells, hotspots, steps, tests.

**Gaps:** Lacks risk scoring; migration path unclear.

**Router tuning:** Add impact x effort; phased plan.

**Upgraded prompt**

Review this code: [paste]. Deliver:  
- Findings by category (correctness, security, perf, clarity);  
- Hotspots with complexity signals;  
- Refactor plan in small, safe steps with tests;  
- Risk/Impact vs Effort matrix (P0/P1/P2);  
- Before/after snippet for 1 key function.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 16) Strict JSON Every Time

**Use when:** Machine‑readable output required.

**Strengths:** Clear schema.

**Gaps:** No parser check; no enum constraints.

**Router tuning:** Include enums & validation note.

**Upgraded prompt**

Return \*\*only valid JSON\*\* for [task]. Schema:  
{  
 "title": "string",  
 "summary": "string",  
 "risks": ["string"],  
 "actions": [ { "owner": "string", "step": "string", "eta": "YYYY-MM-DD" } ],  
 "metrics": ["string"]  
}  
No prose. Validate keys, types, and date format before returning.

――――――――――――――――――――――――

## 17) SOP / Checklist

**Use when:** Repeatable, low‑variance execution.

**Strengths:** Steps + gates + recovery.

**Gaps:** Timing windows; roles not explicit.

**Router tuning:** Add roles & time boxes.

**Upgraded prompt**

Draft a step‑by‑step SOP for [process]. Include:  
- Prereqs & roles;  
- Steps with time boxes;  
- Quality gates with pass/fail checks;  
- Common failure recovery & escalation ladder.  
Mode: [fast/deep]. Output as a checklist.

――――――――――――――――――――――――

## 18) Positioning & ICP

**Use when:** Sharpen message‑market fit.

**Strengths:** ICP, pains, alts, value prop, messages, pitch.

**Gaps:** Jobs vs pains; proof tokens missing.

**Router tuning:** Add JTBD & proof lines.

**Upgraded prompt**

Define positioning for [product]. Provide:  
- ICP traits (firmographic + behavioral);  
- JTBD and top pains (ranked);  
- Alternatives (do‑nothing included);  
- Value proposition (benefit + proof);  
- 3 key messages;  
- 3‑line elevator pitch.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 19) Competitive Teardown

**Use when:** Side‑by‑side clarity.

**Strengths:** Features, UX, pricing, moat, switching costs, objections.

**Gaps:** Buyer role nuance; evidence weak.

**Router tuning:** Add role lens + cite artifacts.

**Upgraded prompt**

Compare [your product] vs [competitor] for [buyer role]. Cover:  
- Features & UX (table);  
- Pricing (typical deal sizes/TCO);  
- Moat & switching costs;  
- Buyer objections + crisp replies;  
- Evidence links (docs, screenshots) if available.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 20) Policy First Draft (Non‑Legal)

**Use when:** First pass policy with clarity.

**Strengths:** Rules, examples, do/don’t, escalation.

**Gaps:** No scope/authority; review cadence missing.

**Router tuning:** Add scope, owner, review cadence.

**Upgraded prompt**

Draft a \*\*non‑legal\*\* first‑pass policy for [topic]. Include:  
- Scope & definitions; policy owner;  
- Rules with examples; do/don’t lists;  
- Compliance checks & escalation path;  
- Exceptions process;  
- Review cadence and change log placeholder;  
- Legal review placeholder.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 21) 7‑Day Learning Plan

**Use when:** Focused upskilling in a week.

**Strengths:** Daily objectives, resources, practice, quiz.

**Gaps:** Entry level varies; no capstone.

**Router tuning:** Add diagnostic + capstone.

**Upgraded prompt**

Build a 7‑day learning plan for [skill/exam]. Include:  
- Day 0 diagnostic (what to skip/focus);  
- Daily objectives, resources (≤3/day), and practice tasks;  
- Daily self‑quiz (5 Qs) with expected answers;  
- Day 7 capstone task with rubric.  
Mode: [fast/deep].

――――――――――――――――――――――――

## 22) Negotiation Prep

**Use when:** Plan the conversation before the room.

**Strengths:** Goals, walk‑away, BATNA, concessions, questions, opening.

**Gaps:** Counter‑plays; objection map missing.

**Router tuning:** Add opponent map + scripts.

**Upgraded prompt**

Create a negotiation brief for [deal]. Include:  
- Goals; walk‑away; BATNA;  
- Concession strategy (give/get);  
- Questions to surface interests;  
- Opening script;  
- Objection map with counters;  
- Opponent/alignment map (roles, power, interests).  
Mode: [fast/deep].

――――――――――――――――――――――――

## 23) Landing Page Copy

**Use when:** Write conversion‑first copy.

**Strengths:** Section list, direct tone.

**Gaps:** Segment nuance; FAQ weak.

**Router tuning:** Add segment option + proof elements.

**Upgraded prompt**

Write a landing page for [offer]. Sections:  
- Headline + subhead (clear promise);  
- Value bullets (3–6) with outcomes;  
- Proof (logos, testimonial lines, metrics);  
- CTA (primary + secondary);  
- FAQ (5–7 Qs).  
Optional: provide a variant for [segment].  
Mode: [fast/deep].

――――――――――――――――――――――――

## 24) Automation Blueprint

**Use when:** Design automations with ROI.

**Strengths:** Triggers, steps, data, errors, alerts, ROI.

**Gaps:** SLAs; run‑costs; auditability.

**Router tuning:** Add SLAs, idempotency, and cost model.

**Upgraded prompt**

Propose automations for [workflow]. Include:  
- Triggers & prerequisites;  
- Steps with systems & data sources;  
- Error handling (retries, dead‑letter, idempotency);  
- Alerts/observability (what, who, channel, thresholds);  
- SLAs & run‑cost model;  
- ROI estimate (baseline vs future, payback).  
Mode: [fast/deep].

――――――――――――――――――――――――

## Bonus: Mini Switches You Can Add Anywhere

* **“Low‑randomness, no lateral riffs.”** For deterministic outputs.
* **“Use a verification pass: compare output vs. constraints, fix before returning.”**
* **“If citing, append a short sources list with titles + links.”**
* **“Label assumptions explicitly if context is thin.”**
* **“Return a ‘How to use this output’ note in one line.”**

――――――――――――――――――――――――

### Final Notes

* Keep the Router Header lean; the power comes from **clear Output Contracts** and **tight constraints**.
* Prefer **JSON** when downstream automation is needed; prefer **skimmable bullets** when humans are the primary consumer.
* If you need extra toughness, combine “adversarial” and “self‑check” lines.

――――――――――――――――――――――――

**Changelog v1.1 (this doc):** Added threat models, self‑check, enum statuses, strict JSON variants, SLAs/costs for automation, and decision‑date/owner fields for memos.