GUT-Check Product Specification

Version 0.3 — Data-Aware, Monetization-Explicit +

---

1. Project Capabilities/Goals

GUT-Check is an evidence-first analytical system designed to evaluate nutrition, supplement, food, and health-adjacent claims under uncertainty. It produces structured, bounded analyses that grade evidence quality, identify assumptions and gaps, assess risk, and clearly state confidence limits.

The primary goal is to improve decision quality while resisting marketing distortion and unsupported health claims.

GUT-Check's competitive advantage is not answers, but disciplined refusal to overstate certainty.

1.1 Research Awareness and Retrieval

A) User story

* "As a user, when I ask GUT-Check a research question, the system shows relevant papers, indicates which are open access, downloads OA PDFs automatically, and shows paywall access options for the rest."

B) Non-goals (important for compliance and scope)

* "No credential-based scraping"
* "No automated login to paywalled publishers"

C) Functional requirements

* Discovery (metadata)
* Relevance ranking
* OA resolution
* OA PDF download
* User library upload/import
* RAG answering with "full text vs abstract-only" labeling

D) Acceptance Criteria

Below are the 5 acceptance criteria, written as testable statements for GUT-Check’s app flows. These are “done means done” checks you (or Claude) can implement as automated tests and/or manual QA scripts.

Acceptance Criteria 1: Research discovery returns ranked works

Given:

* A user submits a GUT-Check request with research.enabled=true and query_text="fungal contamination mushroom cultivation sterilization"
  When:
* The system runs the Research Awareness pipeline (discover -> rank)
  Then:
* The response includes research.works with at least 20 WorkRecord items (or max_ranked if lower)
* Each WorkRecord includes at minimum: title, year, relevance_score, availability_state
* The list is sorted descending by relevance_score
* Each WorkRecord includes a stable id and a source identifier (e.g., openalex)

Acceptance Criteria 2: OA availability classification is present and correct-by-contract

Given:

* At least 10 works in research.works have a DOI
  When:
* The system runs OA resolution (Unpaywall or configured OA resolver)
  Then:
* For each work with a DOI, the response includes oa.is_oa and availability_state
* availability_state is one of: OA_PDF_AVAILABLE, OA_NO_DIRECT_PDF, PAYWALLED_OR_UNKNOWN
* If availability_state == OA_PDF_AVAILABLE, oa.url_for_pdf is non-null
* If availability_state != OA_PDF_AVAILABLE, oa.url_for_pdf is null (or absent, per your schema rules)

Acceptance Criteria 3: OA PDF retrieval happens only when allowed, and produces an ingested library record

Given:

* A work exists with availability_state == OA_PDF_AVAILABLE and oa.url_for_pdf present
* research.settings.download_oa_pdfs == true
  When:
* The system runs the OA fetch + ingest step
  Then:
* The system downloads the PDF successfully (HTTP 200, non-empty file)

* A LibraryDocument record is created in research.library with source == oa_download, text_indexed == true
* The downloaded/ingested doc is linkable back to the WorkRecord (by DOI and/or stored reference)
  And:
* No PDF downloads are attempted for works with availability_state != OA_PDF_AVAILABLE

Acceptance Criteria 4: Paywalled works generate “access options” and no full-text download attempts

Given:

* A work exists with availability_state == PAYWALLED_OR_UNKNOWN
  When:
* The system formats the result set for the user
  Then:
* access_options.publisher_url is present (prefer DOI landing URL when DOI exists)
* access_options.upload_pdf_prompt == true
* access_options.institution_prompt == true (unless disabled in settings)
  And:
* The system does not attempt any automated login, credential use, or paywalled PDF retrieval for that work
* The response clearly labels that this item is “abstract-only” (or equivalent flag) unless user supplies full text later

Acceptance Criteria 5: User-supplied PDFs become first-class sources for answers (RAG correctness)

Given:

* The user uploads a PDF (or imports a library item) that is relevant to the query
  When:
* The system ingests the PDF and the user asks a follow-up question that should be answered from that document
  Then:
* The generated answer includes at least one citation to the ingested LibraryDocument (doc_id and/or internal reference per your JSON schema)
* Retrieved passages are from the user library (not only from abstracts)
* The response includes a provenance indicator for each citation (e.g., source=user_upload vs oa_download vs abstract-only metadata)

Optional “tightening” criteria

A) Caching: If the same DOI is resolved twice, the second run uses cached OA resolution (no duplicate Unpaywall call unless cache expired).
If DOI is missing, availability_state defaults to PAYWALLED_OR_UNKNOWN (or NO_DOI if you choose to add a state), and publisher_url uses landing_url when available.

B) Deduplication: If OpenAlex and Semantic Scholar return the same DOI, the system merges into one WorkRecord.
Research retrieval does not enable prescriptive dosing or treatment recommendations; it only supports evidence assessment.
That keeps the feature from accidentally becoming a “medical advice funnel.
C) Determinism: Given the same query and same cached source responses, output ordering is stable.

1.2 Definitions

* Work, OA, Paywalled/Unknown, User Library, Availability State

1.3 Failure modes

* No DOI
* DOI present but Unpaywall has no OA location
* PDF download fails
* User uploads PDF without DOI
* Duplicate papers across sources

2. Operating Principles

• Evidence before conclusions
• Uncertainty is explicitly surfaced, not hidden
• Context is collected only when it improves analysis
• Users are never required to identify themselves
• Monetization occurs at the insight and tooling level, not via personal data sales

---

3. Scope Boundaries

Included:
• Nutrition and supplement claims
• Functional foods (e.g., beet juice, probiotics)
• Health-adjacent performance and wellness topics
• Media and marketing claim evaluation
• Evidence grading and risk framing

Excluded:
• Medical diagnosis or treatment

• Legal advice
• Prescriptive dosing instructions
• Behavioral persuasion
• Emotional reassurance without evidence

GUT-Check evaluates reliability and applicability, not personal medical suitability.

If only abstracts/metadata are available, confidence_statement must be capped (e.g., no ‘high confidence’ based only on abstracts).”
This makes your “disciplined refusal” enforceable at the code level.

4. Progressive Context Collection Model

GUT-Check explicitly rejects upfront intake forms.

Context is collected progressively and optionally, based on analytical need.

Rules:

1. No question is asked unless it materially improves the current analysis.
2. The reason for each question must be stated in plain language.
3. All questions are optional and skippable.
4. Anonymous use is the default.

Examples of context collected progressively:
• Age range (only if evidence varies by age)
• Sex (only if biologically relevant)
• State or country (regulatory/environmental relevance)
• Source of influence (where the user heard about the claim)
• Motivation (performance, gut health, BP, etc.)
• Use status (considering vs already using)

OPINION: Progressive questioning signals expertise; forms signal extraction.

---

5. Evidence Grading Rubric

All claims are evaluated using the following rubric:

Grade A – High Reliability

• Peer-reviewed studies
• Government or regulatory datasets
• Replicable methods with disclosed limitations

Grade B – Moderate Reliability
• Expert analyses with partial transparency
• Reputable investigative journalism
• Industry reports with disclosed methods

Grade C – Low Reliability
• Anecdotes
• Influencer content
• Marketing claims
• Non-replicated findings

Grade D – Unsupported or Contradicted
• Unsourced claims
• Circular citations
• Assertions contradicted by higher-grade evidence

Evidence grades bound conclusions. No exception.

---

6. Output Contract (JSON-Native)

Every GUT-Check response must include:

• claim_summary
• evidence_inventory (with grades)
• assumptions_and_gaps
• research_awareness (object; may be empty)
• risk_assessment
• confidence_statement
• decision_utility
• next_questions (0–2 progressive questions max)

Outputs are structured JSON first, rendered for humans second.

7. Data Storage and Privacy Model

• Anonymous sessions by default
• No required accounts

• No collection of names, emails, or precise identifiers
• Context stored as categories or ranges where possible
• Identity data (if ever added) kept separate from analytical data

Raw user-level data is never sold.

---

8. Monetization Model (Explicit)

GUT-Check monetizes through:

Allowed:
• Aggregated, de-identified trend insights
• Influence and misinformation mapping
• Evidence-quality gap analysis
• Licensing of the GUT-Check methodology
• Decision-support tools and APIs

Disallowed:
• Sale of raw or identifiable user data
• User-level targeting
• Sponsored conclusions
• Pay-to-upgrade evidence grades

OPINION: Monetizing restraint creates a defensible moat in a distorted market.

---

9. Safety Constraints

Hard constraints:
• No diagnosis
• No treatment plans
• No medical dosing
• No fabricated citations
• No false precision

Soft constraints:
• Avoid absolutist language
• Prefer ranges and conditions
• Flag ambiguity explicitly

---

10. Intended Use

• Individual claim evaluation
• Consumer literacy
• Ethical product development
• Compliance review
• Research prioritization
• Market signal detection

Not intended for:
• Clinical decision-making
• Emergency health guidance
• Advertising copy

---

11. Living Specification

This Product Specification governs:
• Data collection rules
• Monetization limits
• Output guarantees

Goal: Make the Research Awareness and Retrieval feature real and testable at the product level before touching schema details.

All future README files, code, prompts, and databases must conform to this document.