

Found Advantage AI Visibility Platform

Technical Whitepaper v1.0 — An AI-powered, Python-native operating system for SEO, AEO and GEO teams

Oct 18, 2024 · Sally Fong, CTO

1. Executive summary

Found Advantage is a Python platform that runs the daily workflow of a 17-role SEO, AEO and GEO team as a set of supervised AI agents, one dashboard and one shared data model. It diagnoses, prioritizes, drafts and measures; humans approve anything that ships to a live site, a journalist or a client.

Search has split into three surfaces that reward different things. Classic SEO rewards crawlable, relevant, linked pages. Answer engines reward extractable passages. Generative engines (ChatGPT, Gemini, Perplexity, Claude, Google AI Overviews) reward entities that the wider web describes consistently and cites often. Most agencies staff these as separate teams with separate tools, so the same client is audited three times and nobody owns the joined-up picture.

Found Advantage closes that gap with five capabilities:

- **One data spine.** Search Console, GA4, crawl data, rank and backlink feeds, CRM revenue and AI-answer probes land in one warehouse keyed to the same URLs, queries and entities.
- **A Council of agents.** Seventeen specialist agents, each mirroring a human role, read that spine through typed tools and write findings in a common schema.
- **A prioritization engine.** Every finding is scored on impact, severity, effort and confidence, then sorted into Quick Wins, Core Builds, Authority Moats and Experiments.
- **AI visibility measurement.** Scheduled probes ask generative engines the client's money questions and log who gets mentioned, cited and recommended.
- **A revenue-linked scorecard.** Rankings and citations are reported next to leads and pipeline, not instead of them.

What it does not do: it cannot guarantee rankings or inclusion in AI answers, it does not publish without approval, and its AI-visibility numbers are samples of non-deterministic systems, reported with confidence intervals.

2. Problem statement

The unit of visibility has moved from the ranked page to the cited entity, and agency workflows have not moved with it.

Three surfaces, one buyer. A buyer may see a blue link, a featured answer and an AI summary for the same query in one session. Winning one surface while losing the other two leaves revenue on the table.

Fragmented teams. A full-service engagement needs strategy, technical SEO, content, PR, analytics, CRO and AI-visibility specialists. Each uses its own tools and spreadsheets. Hand-offs add weeks, and findings conflict because each team reads different data.

Slow, manual diagnosis. Typical workloads for one mid-size client, per month:

Workflow	Typical manual effort	Main bottleneck
Technical audit and triage	15 to 30 hours	Sorting hundreds of issues into the few that matter
Keyword and intent research	10 to 20 hours	Clustering and mapping demand to pages
Competitor gap analysis	8 to 15 hours	Joining rank, link and content data
Content briefs and refreshes	20 to 40 hours	Research and information-gain checks
AI visibility tracking	5 to 15 hours	Manual prompting with no consistent sample
Reporting	8 to 12 hours	Stitching GSC, GA4 and CRM by hand

These ranges are planning assumptions, not measured benchmarks; teams should replace them with their own time logs.

No measurement for GEO. Most teams cannot answer basic questions: how often an AI engine names the brand for its category, which sources it cites instead, and whether that is improving.

The opportunity. Most of the work above is reading data, matching patterns and drafting. That is work agents do well under supervision, leaving people to judge, approve, build relationships and own outcomes.

3. Design principles

Every engineering choice below traces to ten rules, written so the system can enforce them, not just state them.

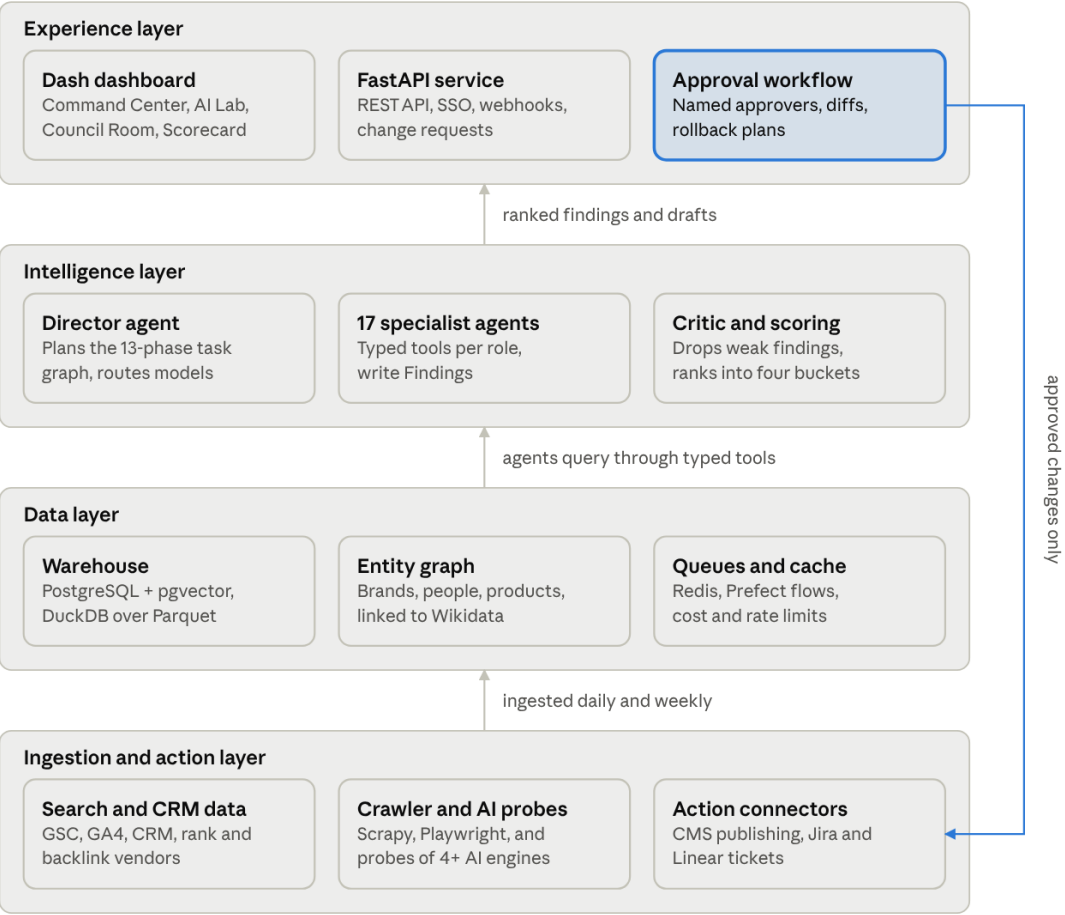
1. **Revenue over rankings.** Every finding carries a `business_link` field naming the metric it moves; findings without one are scored down.
2. **Topics and entities over strings.** Keywords are stored as members of topic clusters and edges in an entity graph, never as a flat list.
3. **Evidence over folklore.** Each finding cites the rows, URLs or probe results it came from. An agent that cannot cite evidence must mark the finding as a hypothesis.
4. **Diagnose before prescribing.** The pipeline runs Business, Demand, Visibility, Authority, Content, Technical, Conversion, Measurement in that order; later agents read earlier agents' outputs.
5. **Few actions, ranked.** The dashboard shows the top ten actions first; the full backlog is one click deeper.
6. **Human in the loop.** Agents may read, analyze and draft freely. Publishing, outreach, redirects, robots and schema changes on live sites need a named approver.
7. **Measurable hypotheses.** Every recommendation ships with a KPI, a baseline, an expected range and a review date.

- 8. **Honest uncertainty.** Confidence is a number from 0 to 100, AI-visibility figures carry intervals, and no output may promise rankings or citations.
- 9. **Python-native and swappable.** Each external API and each LLM sits behind an adapter interface, so vendors can change without touching agents.
- 10. **Moats over hacks.** The prioritization engine rewards assets competitors cannot copy quickly: original data, named experts, earned citations.

4. System architecture

Found Advantage is four Python layers: ingestion pulls data in, the warehouse joins it, agents turn it into ranked findings, and the dashboard routes approved changes back out.

Data flows up into findings; only approved changes flow back out



platform architecture · 4 layers, 12 components

Data only ever moves upward through read-only tools; the single path back to a live site runs through the approval workflow and the action connectors.

Layer	Core Python stack
-------	-------------------

Layer	Core Python stack
Experience	Plotly Dash, Dash Mantine Components, FastAPI, Pydantic, Authlib for SSO
Intelligence	LangGraph, Anthropic and other LLM SDKs, MCP tool servers, scikit-learn, LightGBM, SHAP
Data	PostgreSQL with pgvector, SQLAlchemy, DuckDB, Polars, dbt, Redis
Ingestion and action	Prefect, httpx, Scrapy, Playwright, spaCy, vendor SDKs, CMS APIs

5. Data layer

All agents read one warehouse where every row resolves to four shared keys: `client_id`, `url_id`, `query_id` and `entity_id`. That join is what lets a technical issue, a ranking drop, a lost AI citation and a fall in leads show up as one story.

5.1 Sources and connectors

Domain	Source	Python access	Refresh
Search performance	Google Search Console API, Bing Webmaster API	<code>google-api-python-client</code> , <code>httpx</code>	Daily
Behaviour and conversions	GA4 Data API, server-side events	<code>google-analytics-data</code>	Daily
Revenue	HubSpot, Salesforce, Pipedrive, Shopify	Vendor SDKs, <code>simple-salesforce</code>	Daily
Crawl and render	Own crawler	<code>scrapy</code> , <code>playwright</code>	Weekly, plus on deploy
Page experience	PageSpeed Insights API, CrUX API	<code>httpx</code>	Weekly
Rankings and SERP features	DataForSEO, Semrush or Ahrefs APIs	Adapter per vendor	Daily for tracked set
Backlinks and mentions	Ahrefs, Majestic, Moz, GDELT news	Adapter per vendor	Weekly
Entities	Wikidata, Google Knowledge Graph Search API	<code>SPARQLWrapper</code> , <code>httpx</code>	Weekly
Local	Google Business Profile APIs	<code>google-api-python-client</code>	Daily

Domain	Source	Python access	Refresh
Community	Reddit API, YouTube Data API, review platforms	praw, vendor SDKs	Weekly
AI answers	Anthropic, OpenAI, Google Gemini, Perplexity APIs; browser capture for AI Overviews where terms allow	anthropic, openai, google-genai, httpx, playwright	Daily or weekly by plan

Vendor terms change. Each connector records the terms version it was built against, and the compliance check in section 11 blocks any source whose terms forbid automated access.

5.2 Storage

- **PostgreSQL 16** holds clients, findings, tasks, approvals and the audit log.
- **pgvector** stores embeddings of pages, passages, queries and AI answers for clustering and similarity.
- **DuckDB over Parquet** in object storage (S3, GCS or R2) holds high-volume history: GSC rows, crawl snapshots, probe results. Analysts query it with SQL or Polars.
- **Entity graph** lives in PostgreSQL as node and edge tables, with an optional Neo4j mirror for graph exploration.
- **Redis** handles queues, caching and rate-limit counters.

5.3 Core schema

```
class Finding(BaseModel):
    finding_id: UUID
    client_id: UUID
    agent: AgentRole          # e.g. TECHNICAL_SEO, GEO_STRATEGIST
    phase: Phase              # 1 to 13 of the agency pipeline
    title: str
    evidence: list[EvidenceRef] # rows, URLs, probe ids
    business_link: str | None  # metric this moves
    impact: Literal[1, 2, 3, 4] # low, medium, high, transformational
    severity: float            # 0 to 1
    effort: Literal[1, 2, 3]   # low, medium, high
    time_to_impact: Horizon    # immediate, 30d, 90d, 6_12m
    confidence: int            # 0 to 100
    bucket: Bucket | None     # set by the prioritization engine
    status: Status             # draft, proposed, approved, shipped, measured
    kpi: KPIRef | None
    is_hypothesis: bool
```

Every agent writes this one model. Validation with Pydantic rejects findings with no evidence unless `is_hypothesis` is true.

6. Agent orchestration: the Council

A Director agent plans each engagement as a graph of phase tasks and dispatches them to 17 specialist agents, each mirroring one human role and limited to the tools that role needs.

6.1 The 17 agents

#	Agent	Replicates	Key tools	Writes	Automation level
1	Strategy Director	Chief Digital Strategy Officer	CRM, GA4, business intake	Objectives, ICP, priorities	Drafts; human signs off
2	SEO Strategist	Chief SEO Strategist	GSC, rank feed, clusters	Visibility gaps, cluster targets	High
3	Technical Architect	Technical SEO Architect	Crawler, PSI, CrUX, log files	Ranked technical issues, fix specs	High for diagnosis; fixes need approval
4	AEO Strategist	AEO Strategist	SERP features, passage scorer	Answer gaps, passage rewrites	High
5	GEO Strategist	GEO Strategist	AI probes, citation miner	Citation gaps, source targets	High
6	LLM Visibility Analyst	LLM Visibility Specialist	Probes, brand-description extractor	Entity perception reports	High
7	Entity Architect	Semantic SEO and Knowledge Graph Architect	Entity graph, Wikidata, schema generator	Entity map, JSON-LD	High; schema deploy needs approval
8	Content Director	Content Strategy Director	Inventory, clusters, decay model	Create, update, merge, prune lists	High
9	Editorial Director	Thought Leadership Director	Briefs, interview kits, drafting model	Briefs, drafts, research outlines	Drafts only; human expert edits
10	Digital PR Strategist	Digital PR and Authority Strategist	News feed, journalist DB, pitch drafter	Campaign plans, pitch drafts	Drafts only; human sends
11	Citation Analyst	Link and Citation Authority Analyst	Backlink feeds, mention monitor	Authority gaps, prospect lists	High
12	Local Strategist	Local SEO Strategist	GBP API, citation audit	NAP fixes, review plays	Medium
13	Commerce	Ecommerce SEO	Feeds, facets,	Category and	Medium

#	Agent	Replicates	Key tools	Writes	Automation level
	Strategist	Strategist	product schema	PDP fixes	
14	CRO Specialist	UX and Conversion Specialist	GA4 funnels, page screenshots, vision model	Friction points, test ideas	Medium
15	Measurement Director	Analytics and Measurement Director	Warehouse, attribution model	Scorecards, anomaly alerts	High
16	Competitive Analyst	Competitive Intelligence Analyst	Rank, link, probe data for rivals	Competitive Advantage Map	High
17	Experiment Lead	Growth Experimentation Lead	Test registry, stats library	Hypotheses, test readouts	High for analysis

Local and Commerce agents run only when the business intake flags local or ecommerce revenue.

6.2 How a run works

1. **Intake.** The Strategy Director reads the client brief, connected data and CRM, and writes objectives with target metrics.
2. **Plan.** The Director agent builds a task graph from the 13 pipeline phases, skipping irrelevant agents.
3. **Fan out.** Independent tasks run in parallel (technical crawl, demand research, competitor pulls, AI probes).
4. **Synthesize.** Later-phase agents read earlier findings through the warehouse, not through chat history.
5. **Challenge.** A Critic agent runs the quality-control questions on every finding: evidence or folklore, measurable, materially useful, easy to copy. Weak findings are dropped or demoted to hypotheses.
6. **Prioritize.** The scoring engine (section 8) ranks and buckets the survivors.
7. **Deliver.** The Report agent assembles the client deliverable and CEO summary; humans review before release.

6.3 Implementation

- **Orchestration:** LangGraph for the task graph and checkpointed state; Prefect for scheduled data flows. Either can be swapped for Temporal at scale.
- **Models:** a router assigns a frontier model (for example Claude Opus or Sonnet) to reasoning and drafting, and a small, fast model (for example Claude Haiku) to classification and extraction. Embeddings come from a dedicated embedding model.

- **Tools:** each tool is a typed Python function exposed through tool calling or MCP servers, with Pydantic input and output schemas.
- **Memory:** per-client memory is the warehouse itself plus a short `client_context` document; agents never rely on long chat transcripts.
- **Guardrails:** tool permissions per agent, token and cost budgets per run, output schema validation, and a hard block on write actions without an approval token.

```
@tool(agent=AgentRole.TECHNICAL_SEO, writes=False)
def crawl_issues(client_id: UUID, severity_min: float = 0.5) -> list[CrawlIssue]:
    """Return crawl issues above a severity threshold, with affected URL counts."""
    return warehouse.query(CRAWL_ISSUES_SQL, client_id=client_id, sev=severity_min)

@tool(agent=AgentRole.TECHNICAL_SEO, writes=True, requires_approval=True)
def propose_redirects(client_id: UUID, rules: list[RedirectRule]) -> ChangeRequest:
    """Create a change request; nothing reaches the live site until approved."""
    return changes.create(client_id, kind="redirects", payload=rules)
```

7. Core modules

Twelve modules turn raw data into findings; each lists what it computes, how, and what a human still decides.

7.1 Technical Audit Engine

- **Crawl:** Scrapy for HTML at scale, Playwright for a rendered sample, so raw versus rendered differences expose JavaScript SEO problems (missing links, content, canonicals or meta robots after render).
- **Checks:** status codes, redirect chains, canonical conflicts, robots and noindex conflicts, sitemap coverage versus indexed pages, hreflang reciprocity, pagination, duplicate clusters (SimHash on main content), orphan pages, internal link depth, structured-data validity, Core Web Vitals from CrUX field data.
- **Index reality check:** joins crawl with GSC URL Inspection samples and page indexing reports to separate "crawled but not indexed" from true crawl failures.
- **Triage:** issues are grouped by template, not URL, then scored by affected traffic and revenue. A canonical bug on one template affecting 4,000 product pages outranks 200 missing alt tags.
- **Human decides:** whether and when fixes ship; the module writes developer-ready tickets.

7.2 Demand Intelligence

- Pulls keyword volumes, GSC queries, People Also Ask questions, Reddit and forum questions, and the client's own sales-call and support transcripts where provided.
- Embeds queries and clusters them (HDBSCAN on embeddings, then SERP-overlap merging: two queries sharing 4 or more of the top 10 URLs belong on one page).
- Classifies each cluster by intent and buyer stage (Awareness, Problem Discovery, Education, Evaluation, Comparison, Purchase, Retention) with a small model checked against a labelled sample.
- Estimates value per cluster: volume x expected CTR at a target position x site conversion rate x average order or deal value.

7.3 Competitive Intelligence

- Discovers real search competitors from SERP overlap, not only named rivals, and AI competitors from probe mentions.
- Compares topical coverage per cluster, referring domains, content velocity, SERP feature ownership and AI share of voice.
- Outputs the Competitive Advantage Map: for each cluster, who leads, the gap size, and the cheapest way to close it.

7.4 Content Intelligence

- Inventories every indexable page with traffic, conversions, last update, word count, entities covered and cluster.
- Flags cannibalization (two or more URLs swapping rank for the same cluster), decay (12-week click trend below the cluster trend), thin pages and orphans.
- Recommends one of five actions per page: keep, update, expand, merge (with redirect target) or remove.
- **Information gain check:** compares a draft's facts and entities against the current top results and flags drafts that add nothing new.
- **Brief generator:** outline, questions to answer, entities to cover, internal links, a required original element (data, example, expert quote) and the target passage for answer engines.

7.5 AEO Optimizer

- Splits pages into passages and scores each for extractability: a direct answer in the first 40 to 60 words under a question-shaped heading, self-contained meaning, a definition or list or table where the query implies one, and a cited source.
- Tracks featured snippets, People Also Ask and AI Overview presence for tracked questions.
- Suggests passage rewrites and FAQ blocks, and generates FAQPage, HowTo, Product or Article schema only where the visible content supports it.

7.6 GEO and AI Visibility Lab

The module that has no equivalent in classic SEO tools.

1. **Prompt set.** For each client, 100 to 500 prompts built from the demand clusters: category questions, comparisons, "best X for Y", problem questions and brand questions, written in natural conversational form.
2. **Probes.** Each prompt runs on each tracked engine several times per cycle (default 5) because answers vary run to run. Web-grounded modes are used where available so citations are captured.
3. **Extraction.** A small model extracts brands mentioned, their order, sentiment, whether the client is recommended, and every cited URL.
4. **Metrics.** Mention rate, recommendation rate, citation rate, AI share of voice and cited-source share, each with a confidence interval (section 8).
5. **Source mining.** Cited URLs are clustered by domain and type (review site, news, Reddit, Wikipedia, competitor blog, directory). This becomes the target list for PR and content: the places AI engines already trust for this topic.

6. **Perception audit.** Answers to "What is [brand]?" and "Who is [brand] for?" are compared against the client's own positioning by embedding similarity, exposing wrong or outdated descriptions.

Probe results describe a sample of engine behaviour at a point in time. Engines change models, retrieval and personalization without notice, so the module reports trends, not guarantees.

7.7 Entity and Knowledge Graph Builder

- Extracts entities from the client site and third-party mentions (spaCy plus an LLM pass for disambiguation), and links them to Wikidata IDs where they exist.
- Builds a graph of brand, founders, experts, products, services, industries, locations, research, awards and media mentions.
- **Consistency score:** compares how the brand is described across its site, social profiles, directories, Wikipedia or Wikidata, Crunchbase and press, and flags conflicting names, dates, categories or claims.
- Generates Organization, Person, Product and Service JSON-LD with sameAs links to verified profiles.

7.8 Authority and Digital PR Studio

- Scores backlink and mention gaps against competitors and against the AI-cited source list from 7.6.
- Watches news and social for reactive commentary opportunities in the client's topics.
- Proposes data-led campaigns (surveys, original statistics, indexes, free tools) scored on newsworthiness, link potential and AI citation potential.
- Drafts pitches tailored to each journalist's recent coverage. People send them; the platform never emails journalists automatically.

7.9 Local SEO Module

- Audits Google Business Profile completeness, categories, photos, posts and review velocity and response rate.
- Checks NAP consistency across major directories and flags conflicts.
- Maps service-area demand to location pages.

7.10 Ecommerce Module

- Audits category architecture, faceted navigation (which facet combinations deserve indexable pages, based on demand), product schema, merchant feed errors and review coverage.
- Proposes comparison and buying-guide content tied to category revenue.

7.11 Conversion Analyzer

- Reads GA4 funnels for organic and AI-referral sessions separately.
- Uses a vision model on page screenshots to review message clarity, call-to-action prominence, proof and trust signals above the fold.
- Writes test hypotheses with expected lift ranges for the Experiment Lead.

7.12 Measurement and Attribution

- Builds the KPI scorecard (section 13) from the warehouse.
- Segments AI referral traffic by referrer (chatgpt.com, perplexity.ai, gemini.google.com, copilot.microsoft.com and others) and UTM parameters some engines append.
- Detects anomalies with seasonal baselines (for example STL decomposition) and annotates them with algorithm updates, deploys and campaigns.
- Links organic and AI-referred sessions to CRM outcomes so the dashboard reports pipeline, not just clicks.

8. Scoring models

Five models turn findings and probes into the numbers on the dashboard; all weights are defaults that each agency can tune and must back-test against its own outcomes.

8.1 Priority score

Each finding gets a priority score from 0 to 100:

$$P = 100 \cdot \frac{I}{4} \cdot S \cdot \frac{C}{100} \cdot B \cdot H \cdot E$$

Where I is impact (1 to 4), S is severity (0 to 1), C is confidence (0 to 100), B is 1.0 when a business metric is linked and 0.6 when not, H discounts slow payoffs (1.0 immediate, 0.9 at 30 days, 0.8 at 90 days, 0.65 at 6 to 12 months), and E is effort (1 to 3).

Bucket rules, applied in order:

1. **Experiment:** confidence below 50.
2. **Quick Win:** impact 3 or more and effort 1.
3. **Authority Moat:** horizon 6 to 12 months and the Critic rates it hard to copy.
4. **Core Build:** everything else with impact 3 or more.
5. Remaining low-impact items go to the backlog, hidden by default.

8.2 AI visibility metrics

For a prompt set Q run n times per engine, the mention rate for brand b is the share of answers that name it:

$$\hat{p}_b = \frac{k_b}{n \cdot |Q|}$$

It is reported with a Wilson 95% interval, because small samples of a non-deterministic engine swing widely:

$$\frac{\hat{p} + \frac{z^2}{2N}}{\frac{z^2}{4N^2}} \pm z \sqrt{\frac{\hat{p}(1-\hat{p})}{N}} + \frac{z^2}{4N^2} \Bigg/ \left(1 + \frac{z^2}{N}\right), \quad N = n|Q|, \quad z = 1.96$$

AI share of voice is the client's mentions over all tracked brands' mentions, weighted by position (first mention weight 1.0, second 0.7, third 0.5, later 0.3):

$$\mathrm{SoV}_{\text{client}} = \frac{\sum_a w(\text{pos}_{\text{client}, a})}{\sum_b \sum_a w(\text{pos}_{b, a})}$$

Recommendation rate and citation rate (share of answers citing a client URL) use the same interval method. The dashboard only calls a change real when intervals stop overlapping.

8.3 Passage extractability score

A 0 to 100 score per passage from six checks: answer in the first 60 words (25 points), question-shaped heading (15), self-contained without pronouns pointing outside the passage (15), format matches query type such as list, table or steps (15), a cited source or first-party data point (15), named author with credentials on the page (15). It is a heuristic for prioritizing rewrites, validated by tracking whether rewritten passages gain snippets or citations.

8.4 Citation likelihood model

After roughly three months of probe data, a gradient-boosted classifier (LightGBM) estimates the chance a URL is cited for a cluster from features such as referring domains, topical coverage, extractability, freshness, schema presence, third-party corroboration count and brand mention volume. Its job is ranking which pages to improve first, not predicting any single answer. SHAP values show which features drive each estimate, so the model can be challenged.

8.5 Position scores (1 to 10)

The executive view scores seven areas from 1 to 10 against the top three competitors in the same clusters:

Area	Built from
SEO	Non-branded click share, ranking distribution, cluster coverage
AEO	Snippet and PAA ownership on tracked questions, mean extractability
GEO	AI share of voice, recommendation rate, citation rate
Authority	Referring domains from relevant sites, media mentions, AI-cited source overlap
Content	Cluster coverage, decay share, information-gain pass rate
Technical	Indexable share of valuable URLs, template issue severity, CWV pass rate
Conversion	Organic and AI-referral conversion rate versus site average and benchmarks

Each score is a percentile against those competitors mapped onto 1 to 10, so a 7 means "ahead of most named rivals", not an absolute grade.

9. Dashboard experience

The dashboard is built in Python with Plotly Dash and Dash Mantine Components, served by FastAPI, so the same team that writes agents can ship views. It opens on the ten actions that matter this week, not on charts.

View	Primary user	What it shows	Key actions
Command Center	Account lead, client CEO	Top 10 actions, seven position scores, revenue from organic and AI referral,	Approve, assign, snooze

View	Primary user	What it shows	Key actions
		alerts	
Council Room	Strategists	Live agent runs, findings by phase, Critic verdicts	Rerun an agent, challenge a finding, ask a follow-up
Technical	Technical SEO, developers	Issues by template, affected traffic, index coverage, CWV	Export tickets to Jira or Linear
Demand and Content	Content leads	Cluster map, page actions (keep, update, merge, remove), decay, briefs	Generate brief, approve draft
AI Visibility Lab	GEO strategists	Mention, recommendation and citation rates by engine with intervals; cited-source map; perception gaps	Edit prompt set, add competitor, export source targets
Entity Graph	Semantic SEO	Interactive graph, consistency conflicts, JSON-LD diff	Approve schema change
Authority	PR team	Gap domains, campaign pipeline, pitch drafts, earned mentions	Mark pitch sent, log coverage
Experiments	Growth lead	Test registry, status, readouts with effect sizes	Start, stop, conclude
Scorecard	Management	Weekly and monthly KPIs (section 13)	Schedule report, export PDF

Approval workflow

Every write action becomes a change request with a diff, the evidence behind it, the expected KPI movement and a rollback plan. Approvers are set per client and per change type; for example, redirects need the technical lead, published copy needs the client's subject expert. Approved changes ship through the CMS connector (WordPress, Webflow, Contentful, Shopify) or as developer tickets, and the Measurement agent schedules the follow-up read.

Asking the Council

A chat panel lets users ask questions in plain language ("Why did non-branded clicks fall in September?"). The Director agent answers from the warehouse with linked evidence and can turn the answer into a finding.

10. Workflow replication

The platform automates most of the analysis and drafting in all 13 pipeline phases; the human share concentrates in judgment, relationships and sign-off.

Phase	What agents do	What people still do	Target automation of hours
1 Business diagnosis	Read CRM, GA4 and intake; draft ICP, revenue drivers, objectives	Interview the client, confirm priorities	40%
2 Market and demand	Collect, cluster, classify and value demand	Sanity-check commercial value	85%
3 Competitor intelligence	Find SERP and AI rivals, build Advantage Map	Add market context the data lacks	80%
4 Technical diagnostic	Crawl, render, triage by template, write tickets	Approve fixes, work with developers	75%
5 Content and topical authority	Inventory, decay, cannibalization, briefs	Choose bets, edit drafts, add expertise	60%
6 AEO analysis	Score passages, propose rewrites and schema	Approve copy	75%
7 GEO analysis	Run probes, mine cited sources, audit perception	Interpret shifts, decide responses	80%
8 Entity strategy	Extract and link entities, flag conflicts, generate JSON-LD	Fix off-site profiles, approve schema	65%
9 Authority and PR	Find gaps and angles, draft campaigns and pitches	Build journalist relationships, send pitches	40%
10 Conversion	Funnel analysis, screenshot review, test ideas	Design and build tests	50%
11 Prioritization	Score and bucket every finding	Final call on the top 10	80%
12 Roadmap	Draft 7, 30, 90-day and 12-month plans	Resource and commit	60%
13 Measurement	Build scorecard, detect anomalies, write readouts	Explain results to the client	85%

Automation percentages are design targets for analyst hours, to be validated in the pilot (section 13). They are not claims about outcomes.

The resulting operating model is a smaller senior team supervising the Council: for example, one strategist, one technical lead, one editor and one PR lead can cover the work of a much larger specialist team across more clients, with the platform producing the first draft of nearly everything they review.

11. Security, governance and quality control

Client data stays isolated, every agent action is logged, and nothing reaches a live property without a named human approval.

11.1 Security

- **Tenant isolation:** PostgreSQL row-level security on `client_id`; separate object-storage prefixes and encryption keys per client.
- **Secrets:** OAuth tokens and API keys in a secrets manager (AWS Secrets Manager, GCP Secret Manager or Vault), never in agent prompts or logs.
- **Access:** SSO with SAML or OIDC, role-based permissions mirroring the approval matrix.
- **LLM data handling:** only providers with zero or limited data retention and no training on API inputs; personal data from CRMs is minimized or hashed before any model call.
- **Prompt-injection defence:** crawled pages, reviews and competitor content are treated as untrusted data. They are passed to models in delimited fields, agents that read them have no write tools, and any instruction-like text is stripped and logged.

11.2 Compliance

- Data processing agreements with each client; GDPR, CCPA and the Philippine Data Privacy Act considered for consent and retention.
- Crawling respects robots.txt and rate limits; third-party data sources are used within their terms of service.
- AI-answer capture uses official APIs by default; browser capture runs only where permitted.
- Content generated for publication is reviewed by a human with subject expertise and disclosed in line with client policy.

11.3 Quality control

- **Critic agent** applies the Council's eight questions to every finding before it is shown.
- **Evidence rule** (principle 3) is enforced by schema validation.
- **Golden sets:** labelled samples for intent classification, issue triage and probe extraction, rerun on every model or prompt change; a drop beyond a set tolerance blocks the release.
- **Audit log:** every tool call, model, prompt version, cost and output is stored, so any recommendation can be traced and reproduced.
- **Outcome loop:** shipped changes are measured at 30, 60 and 90 days, and results feed back into confidence calibration so the platform learns which kinds of findings actually work.

12. Deployment, scale and cost

The platform deploys as containers on Kubernetes or a managed equivalent, scales workers by queue depth, and spends most of its variable cost on LLM calls and third-party data.

12.1 Deployment

- **Packaging:** Python 3.12, `uv` for dependencies, Docker images per service (api, dashboard, agent-worker, crawler, scheduler).

- **Runtime:** Kubernetes (EKS, GKE) or a simpler start on Cloud Run or ECS Fargate; KEDA autoscales workers on Redis queue length.
- **Data:** managed PostgreSQL with pgvector, S3 or GCS for Parquet, Redis.
- **Observability:** OpenTelemetry traces across agent runs, Langfuse or similar for LLM traces and cost, Grafana for system metrics, Sentry for errors.
- **CI/CD:** GitHub Actions running unit tests, golden-set evaluations and schema checks before deploy.

12.2 Scale targets

Dimension	Pilot	Production
Clients	5 to 10	200+
URLs crawled per client per week	Up to 50,000	Up to 2 million (sampled rendering)
Tracked queries per client	2,000	50,000
AI prompts per client	100	500
Full Council run time	Under 4 hours	Under 2 hours

12.3 Monthly variable cost per client

An illustrative estimate for a mid-size client, to be replaced with real metering in the pilot:

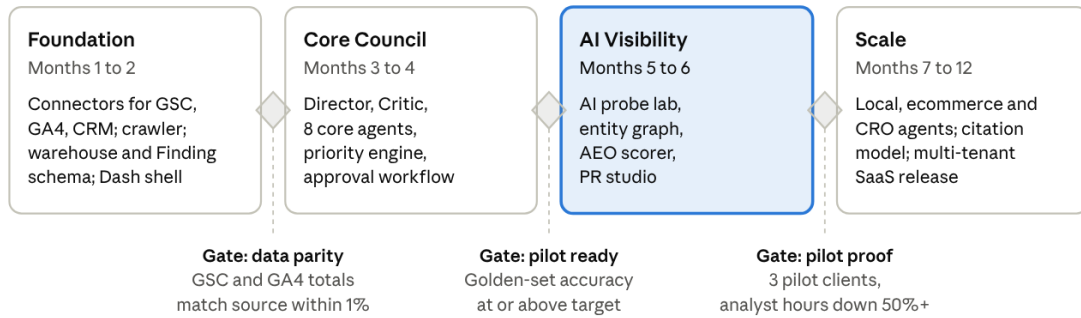
Item	Assumption	Estimated range (USD)
AI visibility probes	300 prompts x 5 runs x 4 engines, weekly	150 to 600
Agent reasoning and drafting	One full run plus weekly refreshes and briefs	100 to 400
Rank, link and SERP data	Vendor API credits	150 to 800
Compute and storage	Crawls, workers, warehouse share	50 to 200
Total		450 to 2,000

Prices for model APIs and data vendors change often; the cost engine reads live rate cards and enforces a budget per client per month. Model routing (small models for extraction, frontier models for reasoning) and prompt caching are the main cost levers.

13. Implementation roadmap and KPIs

The platform ships in four phases over 12 months, and each phase must pass a measurable gate before the next starts.

Four phases take the platform from data spine to SaaS in 12 months



build roadmap · 4 phases, 3 gates, months 1 to 12

AI Visibility is the phase that differentiates the product; it starts only after the core Council proves accurate on labelled data.

13.1 Platform KPIs (is the product working?)

KPI	Definition	Pilot target
Analyst hours saved	Logged hours per client per month versus baseline	50% or more
Finding acceptance rate	Share of top-10 actions approved by strategists	70% or more
Finding hit rate	Share of shipped findings that moved their KPI in the expected direction at 90 days	Tracked; target set after first cohort
Time to first diagnosis	Connect data to delivered executive diagnosis	Under 48 hours
Cost per client	Variable cost per client per month	Within budget in section 12.3

13.2 Client KPI dashboard (is the client winning?)

Area	Weekly	Monthly
SEO	Non-branded clicks and impressions, ranking distribution for tracked clusters, index coverage alerts	Cluster coverage, share of non-branded clicks versus competitors
AEO	Snippet and People Also Ask wins and losses on tracked questions	Mean extractability, zero-click exposure trend
GEO and AI visibility	Mention and citation rates per engine with intervals	AI share of voice versus competitors, cited-source share, perception gaps

Area	Weekly	Monthly
		closed
Authority	New referring domains and mentions	Media citations, expert citations, branded search trend
Business	Organic and AI-referral leads	Qualified pipeline, organic revenue, conversion rate, customer acquisition cost

Management reviews the weekly view for anomalies and the monthly view for direction. Every KPI shows its baseline, its target and the experiments currently aimed at it.

14. Limitations, risks and appendices

The largest risks are moving targets in AI engines, over-trusting generated output, and data access that vendors can withdraw.

Risk	Effect	Mitigation
AI engines change models, retrieval or personalization	Visibility metrics shift for reasons unrelated to client work	Report intervals and trends, annotate known engine changes, keep a fixed control prompt set
Probe answers differ from what real users see	Over- or under-stated visibility	Multiple runs, grounded modes, periodic manual spot checks, AI-referral traffic as a reality check
Hallucinated findings	Wrong recommendations	Evidence rule, Critic agent, golden sets, human approval
Generic AI-written content	Weak pages, possible quality penalties	Information-gain gate, mandatory expert input, editor sign-off
Data vendor terms or pricing change	Gaps in feeds, cost spikes	Adapter layer, at least two vendors per critical feed, budget alerts
Prompt injection from crawled content	Agents misled	Untrusted-data handling, read-only tool scopes for those agents
Over-automation of relationships	Damaged journalist or client trust	Platform never sends outreach; people do
Attribution limits	AI influence on decisions is partly invisible	Report branded search lift and self-reported attribution alongside referrals

Appendix A: Repository layout

```
found-advantage/
  api/           FastAPI app, auth, change requests
  dashboard/     Plotly Dash views
  agents/        Director, Critic, 17 specialists, prompts, tool registry
  connectors/    gsc, ga4, crm, serp, backlinks, ai_engines, gbp, cms
  crawler/       scrapy spiders, playwright renderer
  models/        clustering, intent, extractability, citation model
  scoring/       priority engine, position scores, intervals
  warehouse/     SQL, dbt models, Parquet layout
  flows/         Prefect schedules
  evals/         golden sets and evaluation harness
  infra/         Docker, Kubernetes, Terraform
```

Appendix B: AI visibility probe (simplified)

```
async def run_probe(prompt: Prompt, engine: EngineAdapter, runs: int = 5) ->
list[ProbeResult]:
    results = []
    for i in range(runs):
        answer = await engine.ask(prompt.text, grounded=True)
        parsed = await extractor.parse(answer.text, answer.citations,
                                       brands=prompt.client.tracked_brands)

        results.append(ProbeResult(
            prompt_id=prompt.id, engine=engine.name, run=i,
            mentions=parsed.mentions,           # brand, position, sentiment
            recommended=parsed.recommended,     # brands explicitly recommended
            citations=parsed.cited_urls,
            model_version=answer.model_version,
            captured_at=utcnow(),
        ))
    return results
```

Appendix C: Public API (selected endpoints)

Method	Path	Purpose
POST	/v1/clients	Create a client and connect data sources
POST	/v1/clients/{id}/runs	Start a Council run for chosen phases
GET	/v1/clients/{id}/findings	List findings with filters by bucket, agent, status
POST	/v1/findings/{id}/approve	Approve a finding's change request
GET	/v1/clients/{id}/ai-visibility	Mention, recommendation, citation rates with intervals
GET	/v1/clients/{id}/scorecard	Weekly or monthly KPI scorecard
POST	/v1/webhooks	Subscribe to run, alert and approval events

Appendix D: Glossary

- **SEO:** improving visibility in ranked search results.
- **AEO:** structuring content so answer engines can lift it as a direct answer.
- **GEO:** raising the likelihood that generative AI systems mention, cite or recommend a brand.
- **Entity:** a uniquely identifiable thing (company, person, product) that search and AI systems model.
- **AI share of voice:** a brand's position-weighted share of brand mentions across a prompt set.
- **Information gain:** new facts, data or perspective a page adds beyond what already ranks.
- **Extractability:** how easily a passage can be quoted as a self-contained answer.