PRD: AI Search & Brand Visibility Intelligence Platform

1. Product Overview

Product Name (Working)

AIO Mapper

One-Line Description

A platform that measures, explains, and improves how brands appear, are cited, and are trusted across AI search and generative answer engines.

Problem Statement

Brands have no reliable way to understand:

- Whether they appear in AI-generated answers
- How often they are cited, recommended, or excluded
- Why competitors are referenced instead
- What specific content or authority gaps cause invisibility

Traditional SEO tools do not map to generative retrieval, citation behavior, or answer synthesis logic.

Solution

Provide a system that:

- Simulates real AI questions
- Measures brand presence, absence, and sentiment
- Explains why outcomes occur

- Prescribes concrete, content-level and authority-level fixes

---

2. Goals & Non-Goals

Primary Goals

1. Quantify brand visibility in AI answers
2. Attribute visibility to specific content and authority signals
3. Track change over time
4. Provide actionable remediation guidance

Non-Goals

- Rank tracking for classic SERPs
- Traffic attribution or click modeling
- Black-box “AI score” without explanation

---

3. Target Users & Jobs-to-Be-Done

Primary Users

- Enterprise SEO & Content Leads
- Brand & Comms Teams
- Digital Strategy Directors
- Agency Analytics & Search Teams

Jobs-to-Be-Done

- “Do AI systems recommend us for our category?”
- “Which competitors are cited instead of us?”
- “What content is AI actually pulling from?”
- “How do we improve visibility without guessing?”

---

4. Core Use Cases

1. **Brand Discovery Audit**
    - Is the brand mentioned for category-level prompts?
2. **Competitive AI Share of Voice**
    - Who appears most often in AI answers?
3. **Citation & Source Analysis**
    - Which URLs AI models rely on
4. **Content Gap Detection**
    - What questions AI cannot answer using brand content
5. **Trend Tracking**
    - Visibility before vs. after content changes

---

5. Functional Requirements

5.1 AI Query Simulation Engine

Description
Simulates real user prompts across AI systems.

Requirements

- Prompt library by intent:
    - Informational
    - Comparative
    - Transactional
    - Trust / safety / authority
- Ability to:
    - Run fixed prompts
    - Auto-generate prompts from site taxonomy
- Support for:
    - ChatGPT-style answers
    - Perplexity-style cited answers
    - Search-augmented AI flows

Outputs

- Full answer text
- Brand mentions
- Position in answer
- Citation presence or absence

5.2 Brand & Entity Detection

Description
Identifies how and where brands appear in responses.

Requirements

- Detect:
    - Explicit mentions
    - Implied references
    - Brand variants
- Classify:
    - Primary recommendation
    - Supporting mention
    - Negative / cautionary mention
- Confidence scoring per mention

5.3 AI Citation & Source Mapping

Description
Explains *why* a brand appears or does not.

Requirements

- Extract cited URLs when available
- Map uncited claims to likely source types:
    - Wikipedia
    - Government sites
    - High-authority publishers
    - Brand-owned content
- Identify:

- Missing citations
- Over-reliance on third parties
- Competitor authority dominance

5.4 AI Visibility Scoring Framework

Composite Score (Transparent)

- Presence Rate
- Recommendation Rate
- Citation Rate
- Authority Diversity
- Sentiment Weighting

Rules

- Scores must be explainable
- Each sub-score visible
- No single opaque “AI score”

5.5 Competitive Benchmarking

Description
Shows how visibility compares across brands.

Requirements

- Side-by-side comparison

- Prompt-level breakdowns
- Share of AI answers by brand
- Trend deltas over time

5.6 Prescriptive Insights Engine

Description
Translates findings into actions.

Requirements

- Content recommendations:
    - Missing pages
    - Missing claims
    - Weak explanations
- Authority recommendations:
    - Third-party validation gaps
    - Schema and structure gaps
- Prioritization by:
    - Impact
    - Effort
    - Risk

---

6. Non-Functional Requirements

Performance

- Full audit under 5 minutes per brand
- Incremental refresh for tracking runs

Transparency

- Every metric traceable to:
    - Prompt
    - Answer
    - Evidence

Security & Compliance

- No storage of proprietary prompts without consent
- SOC-ready architecture
- Clear AI usage disclosure

---

7. UX & IA Principles

Core UX Rules

- Always show:
    - The question asked
    - The answer returned
    - Why the result happened
- Avoid:
    - Alarmist language

    - Red/green panic metrics
- Progressive disclosure:
    - High-level summary first
    - Evidence on demand

Key Screens

1. AI Visibility Overview
2. Prompt Results Explorer
3. Brand vs Competitor View
4. Citation Source Breakdown
5. Recommendations Dashboard

---

8. Metrics & Success Criteria

Product Metrics

- Time to insight
- % of users viewing evidence
- Recommendation adoption rate

Customer Outcomes

- Increased brand presence in AI answers
- Increased citation of brand-owned content
- Reduced reliance on third-party summaries

---

9. Risks & Mitigations

Risk

AI outputs change unpredictably

Mitigation

- Aggregate trends, not single runs
- Confidence bands on metrics

Risk

Users misinterpret absence as penalty

Mitigation

- Explicit education on:
    - Coverage vs exclusion
    - Authority vs popularity

---

10. Differentiation vs Existing Tools

Area	This Tool	Profound-Like Tools
Explainability	Full evidence trail	Often abstracted
Scoring	Transparent sub-scores	Composite only
Content Guidance	Page-level fixes	Topic-level
Trust Signals	Explicit authority modeling	Implicit

---

11. Phase Roadmap

Phase 1: Core Visibility

- Prompt simulation
- Brand detection
- Manual competitor input

Phase 2: Authority Intelligence

- Citation mapping
- Source dominance modeling

Phase 3: Continuous Monitoring

- Scheduled re-runs
- Alerting on visibility shifts

---

12. Open Questions

- Should users define their brand entity explicitly?
- How much prompt customization is too much?
- Do we expose raw AI answers or summaries by default?

Key Insight (Important)

You do NOT need to "integrate with ChatGPT" as a UI product.

You need:

- Controlled LLM calls
- Deterministic prompt templates
- Logged outputs
- Repeatable measurement

This is *backend inference*, not chat UX.

Part 2: Phase 1 Scope (Deliberately Constrained)

Phase 1 goal:

> "Can we reliably measure whether a brand appears, is recommended, or is cited when AI answers category-level questions?"

Phase 1 Will Include

- Fixed prompt library
- Single LLM provider
- Brand-presence detection
- Prompt-level outputs
- No competitor auto-discovery
- No sentiment modeling yet

Phase 1 Will NOT Include

- Multiple AI engines

- User-custom prompts
- Real-time monitoring
- Alerting
- Agent orchestration

Part 3: High-Level Architecture Additions

Current Stack (You)

```
Next.js (App Router)
Vercel
Supabase (Postgres + RLS)
Server actions / API routes
```

Phase 1 Additions

```
LLM Inference Layer (server-only)
Prompt Library
AI Answer Store
Brand Presence Analyzer
```

No infra changes. No queues yet. No workers.

Part 4: AI Integration Strategy (Vercel-Safe)

Recommended for Phase 1

OpenAI Responses API (server-only)

Why:

- Deterministic enough

- Supports structured output
- Cheap for batch prompts
- Well supported in Vercel edge or serverless

You do *not* need:

- LangChain
- Agents
- Vector DB
- Tool calling

This keeps your system auditable and explainable.

Example Vercel Pattern

- `/app/api/ai/runPrompt/route.ts`
- Server-only
- No client exposure
- Logged to Supabase

Part 5: Data Model Additions (Supabase)

1. `ai_prompt_templates`

Stores your fixed prompt library.

```
create table ai_prompt_templates (
  id uuid primary key default gen_random_uuid(),
  name text not null,
```

```
  intent text not null, -- informational | comparison | recommendation
  prompt_template text not null,
  is_active boolean default true,
  created_at timestamp default now()
);
```

Example prompt template:

```
"What are the most trusted {category} brands in Canada and why?"
```

---

2. `ai_prompt_runs`

One execution of one prompt.

```
create table ai_prompt_runs (
  id uuid primary key default gen_random_uuid(),
  audit_id uuid references audits(id),
  prompt_id uuid references ai_prompt_templates(id),
  brand_name text not null,
  model text not null,
  raw_response text not null,
  executed_at timestamp default now()
);
```

This is your **ground truth log**.

---

3. `ai_brand_presence`

Normalized analysis output.

```
create table ai_brand_presence (
  id uuid primary key default gen_random_uuid(),
  prompt_run_id uuid references ai_prompt_runs(id),
  brand_detected boolean,
  mention_type text, -- primary | secondary | implied | none
  citation_present boolean,
```

```
  confidence numeric,
  created_at timestamp default now()
);
```

This mirrors your **claim support coverage logic** and will feel familiar.

---

4. Optional (Phase 1.5): `ai_citations`

Only if citations are detectable.

```
create table ai_citations (
  id uuid primary key default gen_random_uuid(),
  prompt_run_id uuid references ai_prompt_runs(id),
  source_url text,
  source_type text, -- wikipedia | publisher | brand | government
  created_at timestamp default now()
);
```

Part 6: API & Server Logic

1. Run AI Visibility Audit (Server Action)

`POST /api/ai/run-visibility`

Input

```
{
  "audit_id": "uuid",
  "brand_name": "PlayNow",
  "category": "online lottery"
}
```

Steps

1. Fetch active prompt templates

2. Interpolate category + brand
3. Run LLM calls sequentially
4. Store raw responses
5. Run brand presence parser
6. Save normalized results

---

2. Brand Presence Parser (Critical Logic)

You already do this conceptually for claims.

Phase 1 rules:

- Exact string match
- Known variants
- Position in answer
- Citation adjacency (if available)

This can be a **pure function**:

```
analyzeBrandPresence(responseText, brandName)
```

No ML yet. Keep it inspectable.

Part 7: UI Integration (Minimal, Non-Disruptive)

Where This Lives in AIO Mapper

Add a **new tab**, not a new product:

“AI Visibility”

Phase 1 UI Components

- Prompt list
- AI answer preview
- Brand presence badge
- Confidence note
- Educational copy

Do **not** mix this with SEO/GEO scores yet.

Part 8: Credit Model Integration

Treat this like a **premium audit extension**.

Suggested cost model:

- 1 AI prompt = X credits
- Phase 1 default run = 5–10 prompts

Reuse your existing `credits` ledger exactly as is.

Part 9: What You Are Explicitly NOT Doing Yet (Correctly)

- No competitor scraping
- No “AI rank”
- No visibility alerts
- No auto-prompt generation

Those come later once signal stability is proven.

Part 10: Phase 1 Success Criteria

You will know this is working when:

- Results are repeatable across runs
- Users understand *why* they appear or not
- Outputs feel grounded, not alarming
- You can explain every score on a call

Final Take (Important)

This is **not a new product**.
It is a **new analysis layer** that fits naturally beside:

- Claim Support Coverage
- Source Transparency
- GEO Readiness

Which is why AIO Mapper can credibly compete with Profound *without* copying it.