

URL Ledger

Product Requirements Document

The system of record for website asset value

Product thesis

Every URL is a business asset with identity, yield, durability, risk, decay, attribution, governance, and machine-readable evidence. URL Ledger turns the website into a measurable asset portfolio - not a pile of pages.

Document type: Internal PRD / product planning specification

Product category: Website Asset Intelligence / URL Asset Ledger

Primary wedge: 45-Day URL Portfolio Repricing Audit

Platform end state: Ledger + ratings engine + policy gate + evidence layer + agent ingress API

Version: v1.0 draft

Intended users: Founder, product, engineering, audit analysts, sales, delivery, and strategic partners

This PRD translates the URL Ledger strategy into product requirements. It does not assume the platform is only for AI Search. AI answers and AI agents are treated as discovery and consumption channels inside a larger system of record for URL assets.

Table of Contents

1. Executive summary
2. Product vision and market context
3. Users, buyers, and jobs to be done
4. Core product architecture
5. MVP scope and non-goals
6. Functional requirements
7. Data model and integrations
8. Ratings and 13-variable scoring
9. Workflow, governance, and agent ingress
10. Dashboard and reporting requirements
11. Security, permissions, and evidence standards
12. Success metrics and release roadmap
13. Open questions and decision log

1. Executive Summary

URL Ledger is a system of record for website asset value. It creates a canonical ledger for every URL, scores the asset across structural and economic variables, turns portfolio decay into a visible management problem, and governs human or agent actions through policy, evidence, and reconciliation.

Strategic principle

Do not position URL Ledger as another AI agent. Position it as the trusted ledger that humans, systems, and agents query before making decisions about URL assets.

Product layer	Purpose	Core output
URL registry	Establish canonical identity, ownership, lineage, and current state for every URL.	Queryable URL asset record
Signal ingestion	Join crawl, search, analytics, CRM, paid, social, email, referral, and manual evidence.	Multi-channel URL observations
Ratings engine	Score each URL across health, decay, risk, fit, ROI, and agent readiness.	Explainable URL rating
Portfolio intelligence	Identify decay, waste, cannibalization, dilution, underinvestment, and protected assets.	Prioritized portfolio map
Policy gate	Control what humans and agents can do to pages based on risk, value, and role.	Approval and action policy
Evidence pack	Capture the proof behind every score, recommendation, and action.	Audit-ready rationale
Reconciliation loop	Compare expected vs actual outcomes after actions are executed.	Learning memory and benchmark input

Primary product outcome

A customer can reconcile a website, identify the highest-value recovery opportunities, govern URL changes, and produce an executive-ready evidence pack without stitching together disconnected dashboards.

Initial commercial product

The first marketable product is the 45-Day URL Portfolio Repricing Audit. The audit is the wedge that installs the URL registry, creates baseline ratings, exposes value-at-risk, and converts the client into an ongoing ledger subscription.

2. Product Vision and Market Context

The web is moving from link-only discovery to fragmented discovery across search, AI answers, paid channels, social, email, referral, direct, marketplaces, internal search, CRM journeys, and browser/enterprise agents. The page remains the atomic unit, but its value is no longer captured by rank and clicks alone.

Old operating model	New operating model
Content is managed as output.	URLs are managed as governed business assets.
Performance is spread across SEO, analytics, CRM, and BI tools.	Every URL has a canonical asset record with joined evidence.
More production is treated as growth.	Portfolio quality, recovery, and capital allocation drive growth.
AI tools write more content.	The ledger decides what should exist, what should change, and what must not be touched.
Dashboards describe symptoms.	The system assigns value, risk, policy, and next-best action.

Product category

Website Asset Intelligence: a management system for measuring, governing, and improving the value of every URL asset across all discovery, attribution, conversion, and agent-consumption channels.

Category wedge

The wedge is not generic SEO. The wedge is content asset repricing: which pages gained value, lost value, became waste, became vulnerable, became protected, became citable, or became structurally invisible after market, algorithm, channel, or site changes.

3. Users, Buyers, and Jobs To Be Done

Persona	Job to be done	Product promise
CFO-minded CMO	Prove which content assets deserve investment and which represent waste.	Portfolio valuation, value-at-risk, recovery potential, and executive reporting.
Head of SEO / Organic Growth	Find structural issues, cannibalization, decay, and recoverable demand.	URL-level scoring, cluster maps, proof sprint backlog, and channel evidence.
RevOps / Demand Gen	Connect URL behavior to pipeline, conversion, and revenue influence.	Attribution joins, assisted-value views, and recovery prioritization by business outcome.
Content / Editorial Lead	Know which pages to refresh, merge, protect, expand, or retire.	Action queue, ownership, protected asset rules, and change history.
Engineering / Web Ops	Prioritize structural fixes by impact instead of ticket noise.	Technical issue clustering, implementation specs, and measurable before/after evidence.
Agency / Multi-site Operator	Govern multiple client portfolios with repeatable scoring and reporting.	Reusable audit workflow, benchmark norms, and portfolio rollups.
AI / Automation Team	Give agents a trusted source layer and policy gate before they act.	Agent-readable URL state, allowed actions, approvals, evidence, and reconciliation.

Core jobs

- Reconcile every URL into a canonical asset registry.
- Determine which URLs are valuable, decaying, duplicative, cannibalizing, risky, or strategically protected.
- Tie URL assets to channel performance, conversion behavior, and economic value.
- Create a ranked action backlog that separates refresh, merge, redirect, retire, protect, and expand actions.
- Govern actions through approval thresholds, evidence requirements, and outcome reconciliation.
- Expose machine-readable URL truth to AI agents and downstream systems.

4. Core Product Architecture

The product should be designed as an asset control plane, not as a content editor. The platform observes state, scores risk and opportunity, recommends actions, enforces policy, and reconciles outcomes.

Layer	Description	MVP requirement
Ingestion layer	Pulls or imports URL inventory, crawl data, GSC, GA4, CRM, paid, email, CMS, and manual evidence.	CSV import plus GSC/GA4-ready schema; connector abstraction.
Canonical URL graph	Normalizes URL identity, canonical tags, redirects, duplicates, merges, splits, clusters, and lineage.	Canonical URL record, cluster ID, redirect/canonical relationships.
Observation store	Stores time-series signals by URL, channel, query, campaign, conversion, and evidence source.	Snapshot tables and observation events.
Ratings engine	Calculates URL-level and cluster-level ratings with explainable factors and confidence scores.	Baseline score, reason codes, evidence links.
Policy engine	Maps page role, rating, risk, and asset value to allowed actions and approval requirements.	Action policy matrix and protected asset flags.
Action queue	Converts findings into refresh, merge, redirect, retire, protect, expand, or	Prioritized backlog with owner, effort, impact, confidence.

Layer	Description	MVP requirement
	investigate tickets.	
Evidence pack	Stores screenshots, exports, charts, rationale, approvals, and before/after snapshots.	Evidence artifacts attached to scores/actions.
Reporting layer	Generates audit reports, quarterly reviews, executive summaries, and proof packs.	Audit report generator and export views.
API / agent ingress	Allows approved systems or agents to query state, request actions, and write back results.	Read API spec and action request workflow.

Transaction model
 Observe -> Score -> Decide -> Act -> Reconcile. The ledger compounds value because every action, decision, and outcome becomes memory.

5. MVP Scope and Non-Goals

MVP scope

Capability	Included in MVP	Acceptance criteria
URL inventory import	Sitemap, crawl export, CMS export, or CSV upload.	Customer can ingest and normalize at least 10,000 URLs.
GSC/GA4 data join	Support import or connector-ready format for page/query and landing page metrics.	URL records show search and behavior observations.
Canonical URL registry	URL ID, canonical URL, page type, status, cluster, owner, published date, updated date.	Every URL has a unique record and current status.
13-variable scoring v1	Score each URL and cluster using configurable heuristics and evidence.	Scorecard outputs rating, risk, confidence, and reason codes.
Portfolio views	Decay, waste, cannibalization, dilution, protected assets, action backlog.	Analyst can generate executive-ready summary.
Audit report export	Generate sample report with findings, backlog, and 90-day roadmap.	Exportable DOCX/PDF or structured report package.
Policy flags	No-touch, high-review, owner-required, agent-safe, manual-only.	Actions display required approval level.
Action queue	Ranked refresh, merge, redirect, retire, protect, expand recommendations.	Backlog can be filtered by impact, effort, confidence, owner.

Non-goals for MVP

- Do not build a content writing tool.
- Do not build a generic SEO recommendations platform.
- Do not promise automatic publishing or autonomous agent execution.
- Do not attempt perfect multi-touch attribution on day one.
- Do not create financial reporting language that implies GAAP compliance.
- Do not require every connector before the audit product can create value.

6. Functional Requirements

ID	Requirement	Priority	Notes
FR-01	System shall create a unique URL_ID for every normalized URL asset.	P0	Core ledger primitive.
FR-02	System shall store canonical, current, historical, redirected,	P0	Required for lineage and reconciliation.

ID	Requirement	Priority	Notes
	duplicate, and orphan states.		
FR-03	System shall classify each URL by page type, intent, role, cluster, owner, and lifecycle state.	P0	Supports scoring and policy.
FR-04	System shall ingest observations from search, analytics, crawl, CRM, paid, email, referral, and manual sources.	P1	MVP can support import templates first.
FR-05	System shall calculate URL and cluster scores across 13 structural variable families.	P0	Explainability required.
FR-06	System shall assign rating labels and confidence tiers to each URL and cluster.	P0	A, B, C, D plus confidence.
FR-07	System shall flag protected assets and enforce high-review/no-touch action rules.	P0	Critical for trust.
FR-08	System shall generate a prioritized action queue ranked by value-at-risk, recovery potential, effort, and confidence.	P0	Primary audit output.
FR-09	System shall store evidence artifacts linked to findings and recommendations.	P0	Audit defensibility.
FR-10	System shall support expected vs actual outcome reconciliation after actions.	P1	Needed for learning memory.
FR-11	System shall export executive-ready reports and analyst-ready working tables.	P0	Sales and delivery requirement.
FR-12	System shall expose read endpoints for approved agents/systems to query URL state.	P1	Agent ingress layer.
FR-13	System shall require action requests to pass policy checks before execution or writeback.	P1	Keeps product differentiated from automation wrappers.

7. Data Model and Integrations

Core objects

Object	Required fields	Purpose
URLAsset	url_id, domain_id, canonical_url, normalized_url, status, page_type, owner, cluster_id, lifecycle_state	Unit of account for the ledger.
URLSnapshot	url_id, date, title, h1, status_code, canonical, indexability, crawl_depth, word_count, schema_status	Point-in-time structural state.
ChannelObservation	url_id, date, channel, impressions, clicks, sessions, conversions, revenue, spend, source_system	Channel-specific performance evidence.
QueryObservation	url_id, query, date, impressions, clicks, avg_position, serp_feature, ai_answer_presence	Search and answer visibility evidence.

Object	Required fields	Purpose
AttributionEvent	url_id, contact_id_hash, account_id_hash, event_type, value, stage, timestamp, source_system	Revenue and pipeline linkage.
Rating	url_id, score, grade, variable_scores, confidence, reason_codes, created_at	Explainable assessment.
PolicyRule	rule_id, scope, trigger, allowed_actions, required_approval, evidence_required	Governance primitive.
ActionRequest	action_id, url_id, action_type, rationale, expected_impact, owner, status, approvals	State transition request.
EvidenceArtifact	artifact_id, linked_object, file_type, source, capture_date, description	Audit-ready support.
OutcomeReconciliation	action_id, observed_delta, measurement_window, confidence, notes	Learning loop and benchmark input.

Integration priority

Priority	Integration / source	Reason
P0	Sitemap, crawl export, CMS export, CSV	Fast path to URL inventory and structural state.
P0	Google Search Console	Search demand, page/query visibility, click and impression baselines.
P0	GA4 or analytics export	Landing page sessions, events, conversions, behavior.
P1	CRM: HubSpot, Salesforce, Zoho, Dynamics	Pipeline and closed-won revenue linkage.
P1	Payments/e-commerce: Stripe, Shopify, WooCommerce	Revenue truth source for commerce or lead-value proxies.
P1	Crawlers: Screaming Frog, Sitebulb, custom crawler	Canonical, redirect, indexability, internal link, depth, and template signals.
P2	Paid platforms and lifecycle email	Cross-channel landing page economics and nurture value.
P2	Data warehouses and BI	Enterprise exports and reconciliation.

8. Ratings and 13-Variable Scoring

The rating is not a black-box SEO score. It is an explainable asset-quality score that combines structural condition, economic value, channel exposure, strategic fit, governance risk, and machine-readiness.

#	Variable family	What it answers
1	Content health	Is the URL still accurate, complete, useful, and current?
2	Content decay	Is value declining over time across traffic, engagement, ranking, conversion, or freshness?
3	Content dilution	Is the portfolio spreading authority and maintenance across too many weak assets?
4	Cannibalization	Are URLs competing against each other for the same intent or demand?
5	Content waste	Does the URL consume maintenance, crawl, or attention without returning value?
6	Content investment	How much cost, effort, expertise, proof, and history is embedded in the asset?
7	Content ROI	Does the URL contribute measurable business value or strategic influence?
8	Performance behavior	How is the URL performing across visibility, traffic, engagement, and conversion?

#	Variable family	What it answers
9	Intent and journey alignment	Does the URL match the real demand state and decision stage?
10	Psychographic and messaging fit	Does the content match user belief, anxiety, proof need, and buying logic?
11	Authority and entity strength	Does the page strengthen brand, topical, author, product, and source trust?
12	Technical and indexation structure	Can systems crawl, index, render, retrieve, and understand the URL?
13	AI citation and agent readiness	Can AI systems cite, summarize, extract, compare, and act through the URL safely?

Rating output

Grade	Meaning	Typical action
A	Healthy, strategic, current, and defensible.	Protect, monitor, expand cautiously.
B	Useful asset with manageable issues.	Selective refresh or targeted improvement.
C	Recoverable but under-managed or structurally weak.	Refresh, merge, reposition, or govern.
D	Weak, redundant, risky, or strategically poor.	Contain, retire, redirect, or exclude from investment.
Protected	High-value or high-risk page requiring special rules.	No-touch or high-approval workflow.

9. Workflow, Governance, and Agent Ingress

The product must govern state transitions. The highest-value actions are not just recommendations; they are proposed changes to the asset portfolio. URL Ledger should track who proposed an action, why, what evidence supported it, what policy applied, who approved it, what changed, and whether the result worked.

Workflow step	Human / system action	Ledger state change
Observe	Import crawl, channel, performance, and evidence signals.	Snapshot and observation records update.
Score	Ratings engine calculates variable scores and confidence.	Rating record created with reason codes.
Decide	Analyst or agent proposes refresh, merge, redirect, retire, protect, or expand action.	ActionRequest created.
Policy check	Policy engine evaluates risk, page role, value, and approval needs.	Action allowed, escalated, blocked, or requires evidence.
Act	Human, CMS workflow, engineering team, or approved agent executes action.	Change log and evidence records update.
Reconcile	System measures expected vs actual results over the agreed window.	OutcomeReconciliation record updates rating memory.

Agent ingress policy requirements

- Agents may read canonical URL state through approved APIs.
- Agents may request actions but cannot execute high-risk changes without policy approval.
- Protected assets require explicit owner review before any writeback action.
- Every agent action request must include rationale, evidence, expected impact, and rollback path.
- The ledger must preserve the distinction between recommendation, approval, execution, and outcome.
- The product should be agent-readable before it becomes agent-writable.

10. Dashboard and Reporting Requirements

View	Primary user	Required components
Executive portfolio view	CMO, CFO, VP Growth	Total URL assets, value-at-risk, recoverable value, grade distribution, top risks, action summary.
URL asset detail	Analyst, SEO, content, RevOps	Canonical state, snapshots, channel observations, rating breakdown, evidence, actions, policy.
Cluster view	SEO, content strategy	Cluster health, cannibalization, dilution, coverage gaps, expand/merge recommendations.
Decay map	Growth and analytics	Decay classes, velocity, half-life, affected channels, recovery priority.
Waste and dilution view	CMO, finance, editorial	Low-value inventory, redundant URLs, maintenance drag, recommended cleanup.
Action queue	Delivery, web ops, content	Action type, owner, impact, effort, confidence, approval status, due date.
Protected asset register	Leadership, legal, brand, RevOps	No-touch URLs, approval rules, evidence requirements, change history.
Agent ingress log	Product, security, automation	Agent/system queries, action requests, blocked actions, approvals, writebacks.
Quarterly Content 10-K	Leadership	Management report on portfolio value, risk, impairment, recovery, and governance.

Reporting exports

- 45-Day URL Portfolio Repricing Audit report.
- Executive readout deck or PDF.
- Action backlog CSV/XLSX export.
- Evidence pack archive.
- Quarterly Content 10-K report.
- Pilot proof pack and case study template.
- Data warehouse export / API response for enterprise systems.

11. Security, Permissions, and Evidence Standards

Permission model

Role	Permissions
Viewer	Read dashboards, reports, and asset records.
Analyst	Create findings, attach evidence, draft recommendations, run scoring.
Owner	Approve actions for assigned URLs or clusters.
Admin	Manage connectors, users, policy rules, scoring configuration, and exports.
Agent / API client	Read approved state and submit action requests within scoped permissions.
External auditor / partner	Restricted read access to evidence packs and reports.

Evidence standards

- Every score should include reason codes that a human can inspect.
- Every high-impact recommendation should include supporting evidence artifacts.
- Every protected asset action should preserve approval history and change rationale.
- Every proof sprint should define baseline, measurement window, expected lift, and observed result.
- Every export should disclose confidence tier and data-source limitations.

- Every automated or agent-assisted recommendation should be labeled as such.

Trust requirement

The product must never feel like a black box. The moat is not that the system produces a recommendation. The moat is that the recommendation has a record, a rating, a policy, an evidence trail, and a measured outcome.

12. Success Metrics and Release Roadmap

Product success metrics

Metric	Definition	Why it matters
Time to first ledger	Hours from access to normalized URL registry.	Measures onboarding speed.
URL coverage	Share of discovered URLs with canonical asset records and basic observations.	Measures completeness.
Scoring coverage	Share of URLs with ratings, confidence, and reason codes.	Measures audit readiness.
Action backlog quality	Share of recommendations with impact, effort, owner, confidence, and evidence.	Measures decision usefulness.
Recovered value tracked	Revenue, conversion, or traffic value recovered after actions.	Measures business proof.
Policy compliance	Share of high-risk actions passing through required approvals.	Measures governance.
Evidence completeness	Share of findings with supporting artifacts.	Measures defensibility.
Audit-to-platform conversion	Share of audits converting to recurring ledger subscriptions.	Measures GTM viability.

Roadmap

Phase	Build focus	Outcome
0-60 days	Audit workflow, URL registry, import templates, ratings v1, report generator.	Manual-assisted audit product that creates sellable evidence.
60-120 days	Dashboard MVP, action queue, policy flags, evidence pack workflow, connector hardening.	Repeatable operating surface for pilot partners.
120-180 days	Outcome reconciliation, CRM/payment integrations, benchmark schema, API read layer.	Proof loop and early benchmark moat.
180-365 days	Policy engine, agent ingress controls, enterprise exports, multi-domain rollups, Quarterly Content 10-K.	System-of-record platform with governance and expansion revenue.
12+ months	Benchmarks, certification, advanced forecasting, agent transaction layer, data warehouse integrations.	Category-defining URL asset intelligence infrastructure.

13. Open Questions and Decision Log

Open product questions

Question	Decision needed
Should the first product UI be audit-operator first or customer-dashboard first?	Decide based on speed to paid pilots.
Should scoring be hardcoded for v1 or configurable by customer segment?	Start with standard rubric plus analyst override.
What minimum data set creates enough value for a paid pilot?	Likely sitemap/crawl + GSC + GA4 + conversion definition.
What is the first revenue truth connector?	Prioritize HubSpot/Salesforce for B2B or Shopify/Stripe for

Question	Decision needed
	commerce.
Should agent ingress ship as API docs first or working endpoints?	API docs and policy model first; limited read endpoints second.
How should benchmarks be anonymized and normalized?	Define peer-set rules before using benchmark claims in sales.
What pages should always be protected by default?	Pricing, legal, comparison, brand proof, high-revenue product/category pages.
How much of the audit report should be auto-generated?	Automate tables and evidence; keep executive narrative human-reviewed.

Immediate next steps

1. Select 3 pilot-friendly website archetypes: B2B SaaS, ecommerce, and agency/multi-site operator.
2. Build the URL asset schema and import templates around the minimum viable data set.
3. Create the audit analyst workspace before building a polished customer dashboard.
4. Implement ratings v1 with reason codes and manual override support.
5. Ship a proof sprint workflow that can show expected vs actual outcomes.
6. Use the first 3-5 audits to harden the 13-variable scoring and benchmark taxonomy.

Final product posture

URL Ledger should become the place where every system asks: What is this URL, what is it worth, what risk does it carry, what can be done to it, and what happened the last time we changed it?

Appendix A: MVP Acceptance Checklist

Area	Acceptance criteria	Status
Inventory	Import at least 10,000 URLs and normalize canonical records.	Planned
Search data	Join page/query observations to URL records.	Planned
Analytics data	Join sessions, events, and conversion observations by landing page.	Planned
Ratings	Generate URL and cluster ratings with reason codes and confidence.	Planned
Backlog	Generate top 25 actions with impact, effort, confidence, and owner.	Planned
Governance	Flag protected assets and approval rules.	Planned
Evidence	Attach supporting evidence to findings and actions.	Planned
Reporting	Export audit report and executive summary.	Planned
Reconciliation	Track expected vs actual result for proof sprint actions.	Planned
API readiness	Document read endpoints and action request contract.	Planned