MCP SERVER | NO CODE | CLOUD HOSTED

Kontent.ai MCP

Audit content types and query structured data from anywhere.

Kontent.ai MCP connects your AI agent directly to Kontent.ai's Delivery API, giving you immediate access to structured content data. You can analyze entire content models, list every available content type and schema, query specific taxonomy groups, find assets in the library, or perform deep searches across your modular repository. Stop guessing what your content looks like; get a complete overview of your project's data structure from any compatible client.

A+ Quality Score 98.33/100

modular-content | api-driven | content-modeling | taxonomy | asset-management

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/kontentai

SCAN TO VIEW

The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

> *The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.*
>
> — Architecture principle

Four Pillars of the Vinkius Runtime

01 — Security by design

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

02 — Built on MCP Fusion

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

03 — Deterministic observability

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

04 — Autonomous operations

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

AES-256	Ed25519	24h TTL	V8 Isolate
Encryption at rest	PKI vault signatures	Ephemeral session keys	Sandboxed execution

One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

```
https://edge.vinkius.com/{token}/mcp
```

Claude · Cursor · VS Code · Windsurf · Grok · Gemini

Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

01 — Ed25519 PKI Vault

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

02 — V8 Isolate Sandboxing

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

03 — SSRF Guard

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

04 — DLP & PII Redaction

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

05 — Cryptographic Audit Trail

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

06 — Honeytoken Trap System

Phantom credentials are injected into isolated environments. If a honeytoken is used outside Vinkius infrastructure, the server is quarantined instantly.

Emergency Kill Switch

EU AI Act Art. 14(1) Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

01 — Server deactivated

The MCP server is immediately taken offline across the entire cluster.

02 — All tokens revoked

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

03 — WebSocket connections killed

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

Control Plane

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Incident Log

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

> Get started at **cloud.vinkius.com** — connect your AI agent in under 60 seconds.

Kontent.ai MCP

10 tools available
Cloud-hosted on Vinkius

Need to understand how complex structured content works? This MCP lets you connect your AI agent directly to Kontent.ai's Delivery API, giving you full visibility into your modular content without ever touching the UI. Instead of sifting through dashboards or guessing what fields exist, you can ask your agent to audit your entire data model, listing all available schemas and their specific elements. If you're building a site that needs structured, reliable content—think product pages with multiple variants, or complex articles requiring author attribution and date ranges—this is the connector for you. You can pull down all necessary metadata, from finding codenames using `list_content_items` to querying entire taxonomy groups. All of this powerful access lives within Vinkius, making it easy to plug Kontent.ai data into your agent workflow, regardless of whether you're working in Cursor or Claude.

Core Capabilities

01 — Audit Content Schemas

Examine the full content types and schemas that define what kind of data exists within your repository.

02 — Query Content Assets

Locate specific images, files, or other media stored in the centralized content library.

03 — Map Content Relationships

Understand how your content is categorized by querying taxonomy groups and associated terms.

04 — Retrieve Specific Content Details

Pull the full, modular details for any single article, product page, or piece of content using its unique codename.

05 — Search Across Repository

Run filtered searches across your entire collection to find relevant items based on specific query parameters.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/kontentai** — connect your AI agent in three steps.

01 Subscribe to the MCP and provide your Project ID from Kontent.ai's settings.

02 If secure access is needed, input your Delivery API Key into the configuration panel.

03 Tell your agent what you need—for example, 'Audit all content types for me,' or 'Find assets related to Q4 promotions.' The MCP executes the query and returns structured data.

The bottom line is that this MCP turns complex, siloed API calls into simple natural language instructions for your agent.

Built For

This is for technical content architects or developers who are tired of manually inspecting CMS dashboards just to figure out a data field name. If you're building an application that needs structured, governed content, this MCP cuts the guesswork.

Content Architect

Uses this MCP to map out and audit all existing content types and taxonomies before a new feature build starts.

Front-End Developer

Calls `get_content_item` or runs `search_kontent_ai` to fetch specific, formatted data needed for dynamic rendering in an application.

Technical Writer/SEO Specialist

Uses the MCP to verify if a required piece of content exists by listing items and checking its associated taxonomy groups.

What Changes When You Connect

01 Stop logging into the Kontent UI just to check a schema. Use `list_content_types` or `get_content_type` to programmatically audit your entire content model with a single prompt.

02 Need to find an image? Instead of navigating through file folders, use `list_content_assets` to query and locate specific media files directly from the MCP.

03 Don't rely on memory for unique identifiers. Use `list_content_items` first to gather all necessary codenames before you attempt to pull down content using `get_content_item`.

04 When building a site that relies heavily on classification, use the MCP to list taxonomy groups and map relationships across your data via `get_taxonomy_group`.

05 Forget running simple keyword searches. Use `search_kontent_ai` to filter content using multiple parameters, dramatically improving search accuracy for agents.

Real-World Applications

The product page needs a new field.

A developer realizes the Product Catalog type is missing a 'Warranty Info' field. Instead of emailing content ops, they ask their agent to run `list_content_types` and inspect the schema using `get_content_type`, immediately identifying the gap for the product owner.

I need all articles mentioning a specific campaign.

A marketing specialist needs to pull every article related to 'Summer Sale.' They prompt their agent with a search query, which uses `search_kontent_ai` and returns a list of relevant codenames. The agent then fetches the full content for review.

How do I find the right image?

A designer needs an asset from last year's campaign but can't remember the file name. They use `list_content_assets` and filter by date range, getting a list of potential images they can then pass to their agent for retrieval.

Mapping out content dependencies.

A data architect needs to know if the 'Article' type depends on a specific taxonomy group. They first use `list_taxonomies` and then drill down with `get_taxonomy_group` to map the exact relationships for their system.

Patterns to Avoid

Treating it like a simple database query

✗ AVOID

Asking the agent, 'What are all my articles?' and expecting a list of every article's full text. This only retrieves metadata.

✓ INSTEAD

First, use `list_content_items` to get codenames. Then, pass one or more codenames to `get_content_item` to retrieve the actual modular content for those specific items.

Assuming all schemas are visible

✗ AVOID

Trying to query a field that doesn't exist on a given content type, causing an error.

✓ INSTEAD

Always start by running `list_content_types` and then use `get_content_type` on the specific schema name. This confirms which fields are actually available before you try to read them.

Over-relying on general keywords

✗ AVOID

Using a broad search query that returns thousands of irrelevant results because it lacks context or filters.

✓ INSTEAD

Use `search_kontent_ai` and pass specific parameters, like filtering by a known taxonomy group ID (obtained via `get_taxonomy_group`) to narrow the results.

The Right Fit

Use this MCP if your core problem is structured data governance or content discovery. If you need to build an application that reads and understands complex, modular schemas (like product variants with multiple required attributes), this connector gives you the necessary visibility. You are modeling the *data structure*, not just generating text.

Do NOT use this MCP if your goal is simple natural language generation or basic web scraping (e.g., 'summarize this webpage'). For those tasks, general purpose text models suffice. If you only need to know what languages are supported, `list_project_languages` works, but if you need the actual content, you must use the full item retrieval tools like `get_content_item`. This MCP is for data engineers and architects; it's about structure first.

The struggle to understand a CMS without developer access.

Today, if you need to know what fields are available on the 'Article' type—say, whether or not an author field is mandatory—you have to log into the Content Management System. You click through tabs, navigate schema views, and copy-paste names just to confirm a data point for a meeting. It's slow, it's prone to human error, and you can't automate the discovery process.

With this MCP, your agent handles that friction instantly. Instead of manual clicks, you prompt: 'Audit all content types.' The system immediately returns a clean, structured list of every schema and its fields. You get the data architecture overview in seconds.

Structured Content Discovery with Kontent.ai MCP

The biggest time sink is verifying content existence or structure. Checking if a product page codename exists, or confirming the required fields for a new article type, requires multiple manual steps across different backend tools.

Now, you can ask your agent to perform these checks in one go. By using `list_content_items` and then cross-referencing that list with schema details from `get_content_type`, you gain immediate data certainty. The entire process becomes a single conversation.

Kontent.ai: 10 Tools for Content Modeling

These tools let you systematically audit your entire content repository structure, from listing every available item to deep-diving into specific data schemas and asset libraries.

#	TOOL	DESCRIPTION
01	`list_content_items`	Lists all available content item codenames, helping you find the unique identifier for any article or page.
02	`get_content_item`	Retrieves the full set of modular data fields and content for a specific item when you know its codename.
03	`list_content_types`	Shows every defined content schema, allowing you to audit what types of content your project supports.
04	`get_content_type`	Provides detailed metadata about a single content type, showing all its required fields and element options.
05	`list_taxonomies`	Lists every taxonomy group defined in your project's structure.
06	`get_taxonomy_group`	Retrieves specific details about a single content classification group, including its terms and hierarchy.
07	`list_project_languages`	Lists all languages that are supported for content within your Kontent.ai project.
08	`list_content_assets`	Queries and lists every media asset, like images or files, stored in the library.
09	`search_kontent_ai`	Performs a flexible search across your entire content delivery repository using various filters.
10	`get_content_type_element`	Gets specific metadata for an element within a type, useful when you need to know the options for a multiple-choice field.

See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

U List the last 10 content items in Kontent.ai

I've retrieved the latest content items. Highlights include 'Homepage' (type: Page), 'Our Mission' (type: Article), and 'Product Catalog' (type: Catalog). Which one should I inspect for you?

U Show the schema for content type 'article'

The 'Article' content type consists of: Title (text), Body (rich text), Author (taxonomy), and Publication Date (date). I can fetch specific articles of this type if you'd like.

U Search for items related to 'Winter Sale'

Searching Delivery API... I found 2 items: 'Winter Sale Banner' and 'Promotion Rules'. I can retrieve the full content for either.

Frequently Asked Questions

01 **How do I check what content types are available using the Kontent.ai MCP?**

Run the `list_content_types` tool first. This will give you a complete list of every schema defined in your project, allowing you to see all possible content structures.

02 **What is the difference between `get_content_item` and `search_kontent_ai`?**

`list_content_items` helps you find a specific item's unique codename. If you have that codename, use `get_content_item` to pull all its data. Use `search_kontent_ai` when you need to find multiple items based on keywords or filters.

03 **Can I list the taxonomy groups using Kontent.ai MCP?**

Yes, use the `list_taxonomies` tool. This gives you an overview of all classification systems in your project, which helps map content relationships.

04 **Does this MCP help me find images or files?**

You can find assets using `list_content_assets`. It queries the content library and returns a list of file metadata, so you know exactly what media is available.

05 **What if I want to see the fields for 'product' type?**

You need to use `get_content_type`. If you provide the content type name, this tool returns all its specific elements and their options, letting you audit the schema.

Go Live in 60 Seconds

Get your connection token from **cloud.vinkius.com**, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

CLIENT	WHERE TO CONFIGURE
Claude AI	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
Cursor	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
VS Code	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add `"kontentai": { "url": "…" }`
Windsurf	MCP Settings → `mcp_settings.json` → Add endpoint URL
ChatGPT	Settings → Tools & plugins → Add MCP server → Paste endpoint
Gemini	Extensions → Add MCP Server → Paste endpoint URL

ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

- Ask ChatGPT
- Ask Claude
- Ask Perplexity
- Ask Gemini
- Ask Grok

READY TO CONNECT

Kontent.ai is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and start building. No SDK. No deployment. Just results.

Start at cloud.vinkius.com →

vinkius.com · support@vinkius.com

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Kontent.ai. All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	Kontent.ai MCP
Server ID	`019d75c2-f15e-737f-b478-5c7a7495bc61`
Platform	Vinkius Cloud for AI Agents
Endpoint	`https://edge.vinkius.com/{token}/mcp`

LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/kontentai.