

MCP SERVER

NO CODE

CLOUD HOSTED

Sitecore MCP

Manage your CMS content without leaving the chat.

Sitecore MCP manages your enterprise CMS directly from any chat window. Use this connector to create, update, delete content items, check item layouts, search deep across templates, or verify workflow status without ever opening the Sitecore Content Editor.

A+ Quality Score 100/100

cms

digital-experience-platform

headless-cms

content-workflow

web-development



The connectivity layer between AI and the world's software.



Vinkius sits between AI and every application. All communication passes through Vinkius Cloud via the Model Context Protocol (MCP) — with governance, observability, and security at every layer.

Your AI Connections Run Through Vinkius Cloud

The world's largest
managed MCP catalog

Vinkius is the connectivity layer where AI connects 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.

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.

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.

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

Encryption at rest

Ed25519

PKI vault signatures

24h TTL

Ephemeral session keys

V8 Isolate

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.

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.

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.

06 — Honeypot Trap System

Phantom credentials are injected into isolated environments. If a honeypot 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.

FinOps

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

Firewall & DLP

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

Agent Activity

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

Tool Health

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

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.

Sitecore MCP

9 tools available

Cloud-hosted on Vinkius

Connect your Sitecore instance to your AI agent and take full control of your complex CMS environment through natural conversation. You can perform core headless operations—like creating new pages or updating field values—without navigating the demanding Sitecore Content Editor GUI. Instead, you simply tell your agent what needs changing. Need to find old content? Your agent searches across templates using keywords and dates. Want to see if a page is ready for publishing? It checks the item's workflow status instantly. When everything else feels like it requires context-switching between your code editor and the CMS backend, this MCP centralizes all that power. Because Vinkius hosts this catalog, you connect once from any compatible client and get instant access to full content lifecycle management.

Core Capabilities

01 — Create new content items

Generate a brand-new item in the Sitecore structure using a specified template and parent location.

02 — Update existing content fields

Modify field values on any current item, specifying which data points need to change.

03 — Find specific content items

Search broadly across the entire CMS content tree using keywords, templates, or date ranges.

04 — Check page status and structure

Verify an item's current workflow state (like Draft or Published) or pull its specific layout details.

05 — Map content hierarchy

List the immediate child items under any given parent node to understand the site structure.

One Click on Vinkius — From Prompt to Execution

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

- 01** First, subscribe to this MCP and provide your Sitecore Base URL along with your Services Client API Key.
- 02** Next, trigger an action request—for example, asking your agent to check the workflow state for a specific page path.
- 03** Your agent sends the necessary calls to Sitecore and returns structured data about item details or content status directly into the chat.

The bottom line is you manage complex digital experiences using simple chat commands instead of multiple logins and GUIs.

Built For

This MCP serves Content Authors who hate context switching, Marketers needing to verify legacy content quickly, and Developers automating test item creation. It's for anyone whose job involves making changes or validating structure within a large-scale enterprise CMS.

Content Author

Uses the MCP to update field values across dozens of pages quickly, and checks workflow states without having to open the main Sitecore client.

Digital Marketer

Runs rapid content searches to locate old campaign assets or verifies if a personalized component is correctly configured in an item's layout.

Front-End Developer

Automates the creation of test items and queries complex Sitecore trees when building headless applications that need mock data.

What Changes When You Connect

-
- 01 Avoid context switching. Instead of jumping between your AI client and the complex Sitecore editor, you manage item creation, updates, and status checks all in one conversation flow.

 - 02 Accelerate content discovery. Use `search_content` to find specific legacy assets or templates by keywords and dates instantly, even if you don't know the exact path.

 - 03 Control the content lifecycle. You can check an item's current status using `get_workflow_state`, verifying whether a page is in Draft or ready for publication before making changes.

 - 04 Build structured data quickly. Instead of manually entering details, use `update_content_item` to push field value changes across multiple pages at once via a JSON object.

 - 05 Map the site structure easily. Running `list_item_children` lets you see exactly what content nodes exist under any parent page, helping developers understand the hierarchy quickly.
-

Real-World Applications

A marketing team needs to verify a campaign's status.

The marketer asks their agent: 'Check the workflow state for the /campaign/SpringSale item.' The agent uses ``get_workflow_state`` and reports that it is still in Review. This prevents them from wasting time trying to edit content that isn't ready.

A content author must update 50 product descriptions.

Instead of logging in repeatedly, the author uses ``update_content_item``, providing a list of item IDs and a JSON payload with new field values. This saves hours compared to manual bulk editing.

A developer needs mock data for testing.

The dev prompts: 'Create a new test article using the Article template under the /content/testing path.' The agent executes ``create_content_item`` and provides the item's new ID, allowing the developer to continue coding immediately.

A user needs to know which templates are available.

The user asks the agent to list all content templates. The agent uses ``list_content_templates`` and provides a clean, categorized list, saving them from having to browse multiple CMS menus.

Patterns to Avoid

Trying to manually navigate paths.**X AVOID**

A user tries to figure out if the 'About Us' page needs a new component by clicking through the Sitecore Content Editor, opening five different child folders, and comparing layout settings in multiple tabs.

✓ INSTEAD

Ask your agent to use ``get_item_layout`` directly. This tool pulls all necessary presentation details for that item into one response, giving you immediate answers without clicks.

Guessing which content templates exist.**X AVOID**

A user assumes a new 'Video Gallery' template exists and tries to create an item using it, only to fail because they don't know the exact template ID or name.

✓ INSTEAD

Always start by running ``list_content_templates``. This confirms what templates are actually available in your site instance before you try to build anything.

Overwriting data without confirmation.**X AVOID**

A user tells the agent to 'change this field' but doesn't specify which item or parent node they mean, risking accidental modification of live production content.

✓ INSTEAD

Be precise. Specify the exact path and use ``get_item_details`` first to confirm you are looking at the correct record before attempting any write operations like ``update_content_item``.

The Right Fit

Use this MCP if your primary job function involves managing content items, verifying CMS structure, or performing CRUD operations within an existing Sitecore environment. This connector is perfect for authors and developers who need to automate routine tasks—like checking workflow status via `get_workflow_state` or creating test nodes using `create_content_item`. Don't use it if you just need to write a simple blog post; that's better handled natively in the CMS. However, don't use it for anything requiring deep knowledge of your site's codebase structure, like debugging C# code—you'd want a dedicated coding assistant tool for that. This MCP is strictly about content and data management within Sitecore.

The friction points in managing enterprise CMS content.

Today, updating content often means logging into the specialized Content Editor. You click through nested folders to find the right item, check its current layout rules, manually adjust field values across multiple pages, and then finally hit 'Submit.' This process is slow, requires dozens of tabs open, and makes it easy to accidentally modify the wrong thing.

With this MCP, you keep everything in your chat window. You tell your agent exactly what needs fixing or updating. The tool handles the complex pathing and data structure behind the scenes. You simply get confirmation that the item was updated or created successfully—no more clicking through a dozen menus.

Sitecore MCP for Content Management Tools

The biggest time sink is context switching. Having to jump from your writing flow into the CMS editor, then back out again, breaks focus and wastes minutes on every single change. You lose momentum just navigating the system.

Now, you keep the workflow in chat. Your agent becomes an invisible copilot that speaks Sitecore's language directly into your conversation. It's instant.

Sitecore: 9 Content Management Tools

These nine tools let you create, modify, find, and audit every aspect of content in a Sitecore instance directly through your agent.

#	TOOL	DESCRIPTION
01	<code>create_content_item</code>	Creates a brand new item within the content tree, requiring you to specify its parent ID, name, and template type.
02	<code>delete_content_item</code>	Permanently deletes an entire Sitecore item. Note that this action cannot be undone.
03	<code>get_item_details</code>	Pulls all current details for a specific Sitecore item using its unique path or ID.
04	<code>get_item_layout</code>	Retrieves the presentation and layout rules that dictate how an item should appear on a page.
05	<code>get_workflow_state</code>	Checks and reports the current status of any item's workflow, such as Draft or Review.
06	<code>list_item_children</code>	Generates a list of all immediate child items residing under a specified parent node.
07	<code>list_content_templates</code>	Provides a list of every available content template that can be used to build new pages or assets.
08	<code>search_content</code>	Performs broad, deep searches for content items across the entire Sitecore repository using keywords and filters.
09	<code>update_content_item</code>	Changes field values on an existing item by passing in a JSON object detailing the name/value pairs to modify.

See It in Action

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

U Can you check the workflow state for the ``/sitecore/content/Home/AboutUs`` item?



The item `/sitecore/content/Home/AboutUs` is currently in the **Draft** workflow state. Would you like me to get its layout presentation details as well?

U List the immediate children under ``/sitecore/content/Home/Products``.



I found 3 child items under the Products node:

1. Laptop X (ID: `110d559f...`, Template: ProductPage)
2. Monitor Y (ID: `220a3...`, Template: ProductPage)
3. Keyboard Z (ID: `330b...`, Template: ProductPage)

Which item would you like me to fetch full details for?

U Create a new item named 'SpringCampaign' using template ``5F22...`` under parent ID ``3D66...``.



The item **'SpringCampaign'** has been successfully created.
New Item ID: `{A1B2C3D4-E5F6-0000...}`.
Path: `/sitecore/content/Home/Campaigns/SpringCampaign`.
Would you like me to update its initial fields with a specific JSON object?

Frequently Asked Questions

01 How do I check the status of a page using the Sitecore MCP?

You use ``get_workflow_state``. Simply give your agent the item's path, and it returns whether that content piece is in Draft, Review, or Published. This saves you from having to navigate the workflow tabs manually.

02 Can I create a brand new page using the Sitecore MCP?

Yes, you use ``create_content_item``. You just need to provide the parent ID, the desired name for the new item, and which template it should use.

03 Is there a way to search old content with Sitecore MCP?

Absolutely. The ``search_content`` tool lets you perform deep searches across your entire CMS using keywords or specific date ranges, even if the item is buried many folders deep.

04 What if I need to update field values on multiple pages?

You use ``update_content_item``. You provide a JSON string detailing the changes and list of items. The MCP handles applying those updates across all specified records in one go.

05 Does Sitecore MCP help with site structure planning?

Yes, you can use ``list_item_children`` to see exactly what nodes are nested under a parent item. This is great for understanding the site's hierarchy before making changes.

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 `"sitecore": { "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

Sitecore 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](https://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 Sitecore. 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	Sitecore MCP
Server ID	019d7608-ab5a-7116-a0f3-98fb420bc6c8
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/sitecore.