MCP SERVER | NO CODE | CLOUD HOSTED

Contentstack MCP for AI Agents

Orchestrate Content Creation and Media Publishing Across Environments

Contentstack lets you manage your headless CMS directly through any AI client connection. You can create, update, and delete content entries using structured JSON payloads, audit schemas, retrieve media metadata, and push live data to development or production environments—all without leaving the chat window.

A+ Quality Score 100/100

digital-experience | content-orchestration | asset-management | workflow-automation | headless-cms

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/contentstack-1

SCAN TO VIEW

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.

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.

Contentstack MCP

10 tools available
Cloud-hosted on Vinkius

Need full control over a headless CMS? This MCP connects your Contentstack account so you can manage complex digital experiences using natural conversation. Instead of logging into a dashboard, you just tell your AI agent what needs doing. You can get structured data by listing and retrieving document rows based on specific content types, or draft new entries with precise JSON attributes. If existing content needs changes, you don't have to click through forms; you can safely update whole entries by overwriting schema blocks. Need to make it live? The MCP triggers the exact publication sequence needed to push CMS data to staging or production environments. It also lets you inspect global schemas and decode field validation rules so developers know exactly what the database expects. When your agent needs media links, you get explicit metadata for assets instead of having to scrape a CDN manually. All this functionality is hosted on Vinkius, giving you one place to connect all your content tools.

Core Capabilities

01 — Create and Draft Content Entries

Generates new CMS draft entries using highly structured JSON payloads.

02 — Update Existing Content Records

Modifies content by safely overwriting schema blocks or substituting draft values in current entries.

03 — Publish to Environments

Triggers automated validation checks, pushing CMS data live to specific deployment environments like development or production.

04 — Inspect Content Schemas

Extracts the structural details of content types and properties used across your entire site.

05 — Manage Media Assets

Retrieves detailed metadata for global files, including original URLs, preventing manual asset scraping.

06 — View Content Listings

Identifies and retrieves all document rows associated with specific content types within the CMS.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/contentstack-1** — connect your AI agent in three steps.

01 Subscribe to this MCP on Vinkius, then provide your Contentstack Stack API Key and Management Token.

02 Directly use your AI client to ask for a content action, like 'List all blog entries for the last quarter' or 'Publish entry XYZ to production'.

03 The MCP executes the necessary calls against your CMS, returning structured data, confirmation status, or the requested metadata.

The bottom line is you control complex content workflows using simple natural language commands through one connection point.

Built For

This MCP is for developers and operations staff who hate context switching. If you're constantly jumping between a CMS UI, a database schema viewer, and your code editor just to publish or audit content, this saves you hours of clicking.

Front-end Developer

You use the MCP to inspect required content type schemas and pull asset URLs directly into your local development environment.

Content Manager

You manage the entire publishing lifecycle, creating drafts and pushing final versions across dev, staging, and production environments without leaving your chat tool.

Product Team Lead

You monitor content model availability and audit field types to plan new digital features before writing a single line of code.

What Changes When You Connect

01 Stop jumping between the CMS UI, your code editor, and deployment dashboards. You manage creation, updates, and publishing status entirely through natural conversation.

02 When you need to audit content type fields or validate rules for a new feature, use `get_schema_details` to pull exact property information without manual database queries.

03 Need to ensure the latest version of content is live? Use `publish_to_environment` to trigger an automated validation check and push data instantly to production.

04 Asset management gets easier. Instead of guessing URLs, use `list_media_assets` or `get_media_asset` to retrieve explicit metadata for every file.

05 The MCP allows you to create draft entries using complex JSON structures with the `create_cms_entry` tool, making content drafting repeatable and precise.

Real-World Applications

Publishing a Site Update

A Digital Ops specialist needs to move a new homepage design from staging to production. Instead of manually verifying the status in three different UIs, they ask their agent to 'Publish entry X to Production'. The MCP executes this via `publish_to_environment`, and they get immediate confirmation.

Developing with Schema Confidence

A front-end developer starts a new product page. They use the agent to run schema inspection, calling `get_schema_details`. The system confirms required fields like SKU type and price format, so they don't build anything that will fail validation later.

Bulk Content Revision

A content manager realizes 50 blog posts need a minor text update. Instead of opening 50 drafts, they use the agent to `update_cms_entry`, passing the new values and targeting all relevant entries with high precision.

Cleaning Up Old Content

A product team needs to clear out old draft content before a migration. They ask the agent to 'Permanently delete app node Y'. The MCP uses `wipe_cms_entry` to ensure the data is gone and quotas are managed.

Patterns to Avoid

Copying UUIDs for API Calls

✗ AVOID

Manually finding an entry's unique ID (UUID) in a list, copying it into the agent prompt, and hoping it works. This is tedious and error-prone.

✓ INSTEAD

Instead of manual copy/pasting, use `list_type_entries` to get a list of available content spaces, then ask the agent to retrieve specific entries by name or type using `get_single_entry`.

Guessing Media Metadata

✗ AVOID

When linking an asset in code, manually guessing the correct CDN path or media ID. This leads to broken images and debugging time.

✓ INSTEAD

Use `list_media_assets` or `get_media_asset`. The agent retrieves the exact structural metadata needed for reliable integration into your front-end build.

Overwriting everything by mistake

✗ AVOID

Running a full content update without specifying which fields are changing, potentially deleting valuable draft data. This is a major risk.

✓ INSTEAD

Always use the `update_cms_entry` tool with explicit instructions on *which* schema blocks to overwrite and *what* values to substitute, ensuring only intended changes take effect.

The Right Fit

Use this MCP if your workflow involves continuous content publishing across multiple environments (dev, staging, production) or if you need deep programmatic access to the underlying CMS schema. It's ideal for teams that write code and manage content simultaneously.

Don't use it if all you need is basic read-only access to a single piece of public information. For simple data retrieval without publishing control, standard API wrappers might suffice. However, if your problem involves auditing schemas (`get_schema_details`), managing asset lifecycles (`list_media_assets`), or guaranteeing the content reaches production reliably (`publish_to_environment`), this MCP is required.

Contentstack MCP for AI Agents: Solving Content Publishing Problems

Currently, publishing a major site update means logging into the CMS dashboard, checking status reports across multiple environments, manually confirming data integrity, and then triggering the deployment sequence. It's slow, it involves copy-pasting UIDs, and every click is a chance for human error.

With this MCP, you simply instruct your agent to 'Publish entry X to production'. The system handles the complex validation checks and triggers the entire publication sequence automatically. You get confirmation that the content data moved successfully, eliminating manual intervention.

Contentstack MCP for AI Agents: Advanced Content Schema Management

Before this tool, figuring out what fields a content model accepted—or if you were allowed to use a specific data type—required digging through

Now, your agent can run schema inspection via `list_global_schemas` . You get an immediate, machine-readable list of all available field types and validation rules, giving developers the

developer documentation or asking the CMS administrator. It was guesswork.

certainty they need to write correct code on the first try.

Contentstack: 10 Tools for CMS Content Orchestration

Use these tools to read content structures, create drafts, retrieve media metadata, and push content changes to any environment.

#	TOOL	DESCRIPTION
01	`create_cms_entry`	Generates new Contentstack drafts using structured JSON payloads.
02	`wipe_cms_entry`	Permanently deletes specific app nodes and live document rows from the CMS.
03	`get_media_asset`	Retrieves detailed structural matching for explicit media IDs.
04	`get_schema_details`	Performs a deep extraction of properties that define active content fields.
05	`get_single_entry`	Retrieves specific content entries by their unique UUIDs for tracing or viewing.
06	`list_media_assets`	Scans deep arrays to list all available media assets, mitigating picture limits.
07	`list_global_schemas`	Lists and enumerates every structured content rule attached across the platform.
08	`list_type_entries`	Identifies all bounded document spaces within your CMS schemas for review.
09	`publish_to_environment`	Sends an automated validation check, pushing content data live to a designated environment (dev, prod, staging).
10	`update_cms_entry`	Safely modifies global web CRM boundaries by substituting new draft values into existing entries.

See It in Action

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

U I need to update the 'About Us' page entry. I changed the main text and added a new team photo; can you update it?

Content Update Status:

- **Entry ID:** `entry-9001` (Type: AboutUs)
- **Status:** Processing...
- **Changes Applied:** Main Text Block overwritten. New Asset (`asset_782`) linked to 'Team Photo' field.
- **Next Step:** Publishing the changes now... ✅

The update is complete and pushed live to the staging environment.

U What are all the current available content types in my system? I need to know what fields exist.

I found 4 global schemas:

1. **Homepage:** Requires Text (Hero Title), File Reference (Main Image), and List (Feature Blocks).
2. **Blog Post:** Needs Text (Body), Symbol (SKU), and Date.
3. **Product Schema:** Includes Number (Price) and Boolean (Is Available).
4. **Contact Page:** Simple text fields only.

You can use these schema details to build your components.

U List the media assets I have for the 'product' category so I can get their IDs.

Here are 5 high-priority media assets available in your global library:

MEDIA ID	ASSET NAME	ORIGINAL URL EXCERPT	SIZE
`media_123`	Hero Banner V3	/assets/banner.jpg	4MB
`media_900`	Product Shot A	/assets/shotA.png	1.2MB
`media_782`	Team Portrait	/assets/team.webp	850KB
...	(3 more assets listed)		

Frequently Asked Questions

01 **How can I use Contentstack MCP to publish content changes without going into the CMS?**

You tell your agent exactly which entry and environment you want updated. The MCP handles the necessary validation checks and pushes the data live, confirming when it's done across development or production.

02 **I need to know what fields I can use for a new content type; how does Contentstack MCP help?**

The MCP lets you inspect global schemas. You run schema inspection to decode the exact field types and validation rules available, so your development is precise from day one.

03 **Can I use Contentstack MCP to draft content entries with specific data formats?**

Yes. You can generate new drafts using purely formatted JSON attributes via the `create_cms_entry` tool, which guarantees your data payload matches what the CMS expects.

04 **How do I get media assets and URLs from Contentstack MCP?**

You use the agent to retrieve explicit media metadata. It provides the original Contentstack URLs for every asset ID you specify, eliminating manual CDN scraping.

05 **Does Contentstack MCP help me audit old or unused content in my CMS?**

Yes. You can use the `list_type_entries` tool to identify all existing document spaces and review which entries exist for a given schema, helping you clean up quotas.

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 `"contentstack-1": { "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

Contentstack 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 Contentstack. 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	Contentstack MCP
Server ID	`019d757b-4710-72df-b384-94073a8fc1a6`
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/contentstack-1.