MCP SERVER | NO CODE | CLOUD HOSTED

Kentico (CMS & DXP) MCP

Manage Content and System Objects via Conversation

Kentico Xperience MCP lets your AI client manage an entire Digital Experience Platform and CMS using natural conversation. You can pull documents by alias path, update site pages, list global user accounts, or extract data from custom tables—all without writing API calls. It gives you full control over content lifecycle management and system settings inside Kentico Xperience.

A+ Quality Score 100/100

digital-experience | dxp | web-content | site-management | enterprise-cms | content-orchestration

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/kentico-cms-dxp

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.

Kentico (CMS & DXP) MCP

10 tools available
Cloud-hosted on Vinkius

Need to manage a complex CMS like Kentico Xperience but hate diving into REST documentation? This MCP connects your agent directly to the platform's core functions, letting you handle content orchestration with simple chat commands. You can ask it to retrieve specific site documents based on their path or culture, update template settings, and even audit user roles across the entire system. It’s like having a CMS super-user sitting in your agent, ready to execute complex tasks instantly. Because Vinkius hosts this connection, you get immediate access to all these core tools from any compatible client, making platform management feel natural again.

Core Capabilities

01 — Manage Site Content

Fetch and modify site pages or documents by their unique paths, handling different languages and sites automatically.

02 — Control System Settings

Create, update, or delete core system objects like templates, roles, and general platform settings.

03 — Audit User Accounts

List all global users and pull detailed profiles, including their associated permissions and metadata.

04 — Analyze Custom Data

Pull tabular data from custom tables to monitor business-specific records that aren't part of the core CMS structure.

05 — Inspect Data Structure

See what types of objects and properties Kentico uses, helping you understand the underlying schema without guesswork.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/kentico-cms-dxp** — connect your AI agent in three steps.

01 Subscribe to this MCP on Vinkius and provide your required Kentico REST URL and authentication header.

02 Your AI client connects using those credentials, gaining full read/write access to the platform's data model.

03 You simply talk to your agent: 'Update the About Us page document name to Our Team,' and it runs the necessary API calls in the background.

The bottom line is you get immediate conversational control over complex, structured CMS operations that normally require multiple manual steps or dedicated code blocks.

Built For

Content Architects and System Administrators are the primary users here. If your job involves maintaining consistency across dozens of site versions or managing user permissions across a large enterprise, this MCP saves hours of API guesswork.

Digital Experience Architect

Uses it to audit document properties and custom table schemas across multiple sites and cultures quickly.

System Administrator

Manages user accounts, roles, and system-level objects for the entire Kentico instance without logging into an administrative panel.

Full-Stack Developer

Tests specific REST service queries or needs to inspect a niche system object type through natural conversation instead of building boilerplate API wrappers.

What Changes When You Connect

01 Content updates become instant. Instead of manually calling APIs to find the right page, you just tell your agent to use `get_site_document` or `update_site_document`, and it handles the pathing for you.

02 System maintenance is simplified. You can list all system objects using `list_system_objects` and then immediately manage them by running `create_system_object` or `delete_system_object` in a single chat session.

03 User auditing gets precise. Use `list_users` to see everyone, then use `get_user` to pull specific role details for compliance checks—no more clicking through admin dashboards.

04 Custom data is exposed. Need metrics from an internal registry? Run `list_custom_table_rows` and get the raw table data directly into your agent's context for analysis.

05 Schema understanding improves fast. If you're unsure how a property works, you can inspect the object types using the schema visibility tool to figure out what fields are available.

Real-World Applications

Rolling back an outdated landing page

A content architect notices a key marketing page has incorrect metadata. They ask their agent, 'Use `get_site_document` for the current About Us page in English.' The agent fetches the data structure, and they immediately use `update_site_document` to correct the title field before publishing.

Onboarding a new regional site

A developer needs to set up an entirely new country website. They ask their agent to 'List all system objects,' identify required templates, and then use `create_system_object` for the initial setup of roles and permissions.

Investigating a suspicious user role

An admin suspects an account has too much access. They run `list_users` to find the ID, then use `get_user` on that specific ID to pull all associated roles and metadata for review.

Pulling inventory data from a custom source

The team needs current stock numbers not stored in the main CMS. They tell their agent to 'List rows from the Inventory table,' and it returns structured, actionable data for immediate reporting.

Patterns to Avoid

Assuming global API access

✗ AVOID

Trying to use your AI client to modify a record that isn't specifically managed by Kentico Xperience (e.g., updating an external CRM field).

✓ INSTEAD

This MCP only controls the CMS layer. To fix it, you need a dedicated connection to that third-party system or another type of integration tool.

Ignoring required authentication details

✗ AVOID

Attempting to run `update_system_object` without providing the correct Basic base64 header credentials, resulting in an immediate 401 error.

✓ INSTEAD

Always verify your Kentico REST URL and Authentication Header during setup. If the connection fails, check those two items first.

Over-relying on simple listing tools

✗ AVOID

Running `list_system_objects` and thinking that just because it lists roles, your agent can arbitrarily change them without knowing the specific role ID or type.

✓ INSTEAD

Use `get_single_object` first to pull detailed metadata about a system object. This confirms the structure before attempting to use `update_system_object`.

The Right Fit

Use this MCP if your core pain point centers around managing content, roles, users, or custom data *inside* a Kentico Xperience CMS instance. If you need to retrieve the structure of documents by alias path (`get_site_document`), audit user permissions (`list_users` , `get_user`), or modify templates/roles (`update_system_object`), this is your tool.

Don't use it if you are trying to build a full accounting ledger, access

data from an external payment gateway, or interact with systems that aren't connected via the Kentico REST service. For those cases, you need a different category of connector entirely.

CMS administration used to be a mess of clicks and dropdowns.

Today, updating a major page involves logging into the CMS, navigating through site structures, finding the right document ID, checking which culture/site version it belongs to, then manually inputting the new metadata—all while hoping you don't overwrite something critical.

With this MCP, that whole process is gone. You just tell your agent what you want changed, and it executes `update_site_document` across all necessary parameters automatically. It handles the complexity so you only deal with the outcome.

Control System Object Management with create_system_object

Before, adding a new type of template or role meant manually making multiple API calls to ensure all dependencies were met and that you used the correct structured REST mappings. It was slow and prone to human error.

Now, your agent manages this complexity for you. You talk about the desired object, and it uses `create_system_object` to build it out correctly from scratch. The platform structure adapts instantly.

Kentico (CMS & DXP) - 10 Tools

These tools allow your AI client to perform specific operations within the Kentico Xperience environment, from reading documents to modifying core system settings.

#	TOOL	DESCRIPTION
01	`create_system_object`	Builds and adds a brand new core platform setting or structure within Kentico.
02	`delete_system_object`	Removes an existing system object from the CMS, maintaining data integrity checks.
03	`get_site_document`	Retrieves a specific website document based on its path in the Kentico CMS.
04	`get_single_object`	Fetches the details of one specific, general system object using its unique ID number.
05	`get_user`	Pulls a detailed profile and role list for any specified Kentico user account.
06	`list_custom_table_rows`	Displays all the rows of data stored in a specific, custom-built business table.
07	`list_system_objects`	Lists different types of system objects available across the entire Kentico platform (like roles or templates).
08	`list_users`	Generates a full list of every user account registered in the global Kentico directory.
09	`update_site_document`	Modifies the content or metadata of an existing document on a live website page.
10	`update_system_object`	Changes settings or properties for a core system object already in use by the platform.

See It in Action

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

U Get the 'Home' page document from site 'CorporateSite' in 'en-us' culture

Retrieving document '/Home'... I've fetched the page structure. It includes properties like DocumentName: 'Home', DocumentMenuCaption: 'Welcome', and the associated page template ID. Would you like to see the full content fields?

U Update the DocumentName of '/About-Us' to 'Our Team' on currentsite

Updating document '/About-Us'... The name has been successfully changed to 'Our Team'. I've also updated the associated metadata. This change is now active in the Kentico database.

U List all rows from the custom table 'MyCompany.Inventory'

I've retrieved 12 entries from 'MyCompany.Inventory'. Highlights include items 'SKU-101' (Stock: 45), 'SKU-105' (Stock: 12), and 'SKU-108' (Stock: 0). Would you like to see the detailed audit logs for these items?

Frequently Asked Questions

01 **How do I use the Kentico (CMS & DXP) MCP to get a specific page?**

You use `get_site_document`. Just tell your agent the path and which site or culture you need. It pulls the full document structure, not just a simple snippet.

02 **Can I list all users using the Kentico (CMS & DXP) MCP?**

Yes, use `list_users`. This tool generates a comprehensive list of every global user registered in your CMS instance for quick auditing purposes.

03 **What if I need to update content on multiple sites? Use Kentico (CMS & DXP) MCP.**

You instruct your agent to perform the bulk action. It coordinates updates using `update_site_document` across different site aliases and cultures as needed.

04 **Is the data from custom tables accessible via Kentico (CMS & DXP) MCP?**

Yes, use `list_custom_table_rows`. This tool lets you pull raw tabular records from business-specific tables that live outside the main document structure.

05 **Does this MCP handle deleting content? Kentico (CMS & DXP) MCP?**

Yes. You can use `delete_system_object` or other functions to irreversibly remove unused system objects, keeping your repository clean and optimized.

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 `"kentico-cms-dxp": { "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

Kentico (CMS & DXP) 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 Kentico (CMS & DXP). 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	Kentico (CMS & DXP) MCP
Server ID	`019d75c0-61c6-701b-9af0-c3f6c4eb4f46`
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/kentico-cms-dxp.