MCP SERVER | NO CODE | CLOUD HOSTED

ButterCMS MCP for AI Agents

Extracting Structured Data and Content Taxonomy from Publishing Sites

ButterCMS MCP connects any AI agent to your entire publishing instance, letting you extract rich text and structured data directly from your Headless CMS. Use it to search blog posts by keywords, map content collections, or analyze internal taxonomy without ever opening the browser console.

F Quality Score 8.79/100

headless-cms | content-api | taxonomy | data-extraction | knowledge-base

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

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.

ButterCMS MCP

10 tools available
Cloud-hosted on Vinkius

This MCP lets your AI agent talk directly to your ButterCMS backend. Instead of manually navigating through dashboards or writing complex API calls, you simply ask for what you need—whether it's finding all articles about a specific topic or mapping out how different content types connect. Your agent reads structured data and rich text assets across your entire publishing site. It can pull detailed information from blog posts by slug, dig into the internal structure of tags and categories, and even enumerate complex custom page models spanning multiple layers of your website. Everything you need to analyze your content warehouse is accessible through natural conversation. When paired with Vinkius, you connect once and get access to this CMS intelligence alongside thousands of other data sources.

Core Capabilities

01 — Scan Blog Content by Topic

Your agent searches the blog posts using keywords or structures to gather specific knowledge from your site's published articles.

02 — Map CMS Taxonomies

You can analyze all configured tags, authors, and categories across the entire system to understand how content is grouped and structured.

03 — List Content Collections

The MCP enumerates all globally attached structured rules and custom page models, giving you a map of your site's data structure.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your specific ButterCMS API Token from your project security settings.

02 Your AI agent connects using that token, giving it direct read access to all structured content within your CMS instance.

03 You ask a question—like 'Show me all articles about X' or 'What are our core categories?'—and the MCP returns clean, usable data directly into your chat window.

The bottom line is you bypass manual database queries and get immediate access to deep content insights right where your AI agent works.

Built For

This MCP is for content experts, marketing strategists, and developers who spend time mapping out complex websites. If you ever feel like you're spending hours clicking between CMS sections just to gather metadata, this tool saves your sanity.

SEO Specialist

You use the MCP to evaluate existing tag structures or locate specific authors' articles quickly so you can recommend backlinking strategies and content improvements.

Content Writer

Need a quick audit? You run rapid scans using this MCP to verify if an article covering a target keyword already exists in the official knowledge base, preventing redundant work.

Headless Engineer

You debug component structures by inspecting nested JSON output to figure out why specific custom pages aren't rendering correctly across your website layers.

What Changes When You Connect

01 Instead of manually checking multiple CMS tabs, you can run a deep search using 'search_blog_posts' to gather knowledge across your entire site in one prompt.

02 You get immediate clarity on content structure; use 'list_butter_categories' and 'list_butter_tags' together to map out how every article is categorized and tagged.

03 Bypass the need for complex coding. Use 'list_custom_pages' to inspect raw page models, helping engineers debug why certain custom data isn't rendering right.

04 Quickly audit your content base by running 'list_butter_authors' to see who wrote what, which is crucial when proposing backlinking efforts to improve SEO.

05 Understand the site map instantly. Running 'list_global_collections' shows you every structured rule attached to your content, giving you a full view of your data architecture.

Real-World Applications

Auditing Content Gaps for SEO

A marketing manager needs to know if their competitor's topic is covered. They ask the agent to 'search_blog_posts' using a niche keyword phrase, immediately identifying relevant articles and content gaps.

Mapping Site Architecture for Devs

A headless engineer needs to verify if a specific custom page template is pulling its data correctly. They use 'list_custom_pages' to inspect the raw JSON output, confirming component hits and nested structures.

Analyzing Content Ownership for PR

A PR team needs to compile a list of all recent articles written by a specific author. They use 'list_butter_authors' combined with 'get_post_details' to quickly build a resource page.

Understanding Content Relationships

A content strategist wants to know the core pillars of their site. They ask the agent to list all categories and tags using 'list_butter_categories' and 'list_butter_tags', creating an instant, usable taxonomy map.

Patterns to Avoid

Treating CMS like a database

✗ AVOID

Trying to copy-paste Markdown text or run simple SQL queries against the content. This fails because you need structured data extraction, not raw file dumps.

✓ INSTEAD

Instead of writing code, ask your agent to use 'list_blog_posts' and then follow up by asking for specific details on a post slug using 'get_post_details'. The MCP handles the structure.

Ignoring content relationships

✗ AVOID

Assuming all related data is in one place. You might only see posts but miss out on the actual organizational units like tags or collections.

✓ INSTEAD

Always check the taxonomy first. Run 'list_butter_categories' and then run 'list_butter_tags'. This ensures you capture both the high-level groupings and the granular metadata.

Overlooking custom page data

✗ AVOID

Only querying standard blog posts, missing crucial content housed in unique landing pages or complex collections.

✓ INSTEAD

Always run 'list_global_collections' to see all potential structured areas. If a collection looks promising, use 'list_custom_pages' to inspect its underlying data model.

The Right Fit

Use this MCP if your primary need is deep content intelligence—specifically extracting metadata, mapping taxonomy relationships, or auditing the structure of highly complex, headless CMS environments. You should use it when you need structured JSON output that goes beyond simple text searches. Don't use it if you just need to draft a new blog post; for that, you need a drafting tool. If your goal is merely to retrieve an API key or check system status, this MCP won't help—you need direct developer credentials access instead.

ButterCMS MCP: Analyzing Publishing Site Taxonomy with AI Agents

Today, gathering a complete picture of your content is painful. You have to jump between the blog list, the tag editor, and the category sidebar—manually checking each section just to map out if 'AI' is tagged as both a Category and a Tag, or if a specific author has written about it multiple times across different collection types.

With this MCP, you ask your agent to perform an audit. It automatically runs checks like listing all defined tags using the 'list_butter_tags' tool and enumerating categories via 'list_butter_categories'. The result is one clean report showing every relationship, saving hours of manual cross-referencing.

ButterCMS MCP: Debugging Custom Page Models with AI Agents

When custom pages break or display unexpected data in a live environment, the usual process is debugging deep within a development console. You have to guess which internal array holds the correct JSON structure for the component you're trying to fix.

This MCP changes that by letting you use 'list_custom_pages'. It inspects the raw page models and returns the explicit structured JSON objects, allowing engineers to pinpoint exactly where a data hit is failing without guesswork.

ButterCMS: 10 Tools for Content Metadata and Taxonomy Mapping

Use these tools to gather structured data, list authors, search blog posts by keywords, or map out every category and collection in your CMS.

#	TOOL	DESCRIPTION
01	search_blog_posts	Find articles and properties that contain specific keywords or topics within your blog posts.
02	search_collection_field	Identify specific arrays of filtered collections to understand which data groups are active on your site.
03	get_page_layout	Retrieve the exact structural matching verifying explicit UI routing
04	get_post_details	Fetch the full details and logging information for any specific post slug you provide.
05	list_butter_authors	Get a list of all authors who have published content through the CMS.
06	list_butter_categories	Analyze and retrieve every single defined category used across your entire site structure.
07	list_global_collections	Enumerate all structured content rules attached globally, showing what types of items are tracked.
08	list_custom_pages	Inspect and list deep internal arrays containing raw data models for custom-built pages.
09	list_blog_posts	List all available blog posts, identifying the general content routing spaces within the CMS limits.
10	list_butter_tags	Retrieve a comprehensive list of all global tags applied to content items for taxonomy mapping.

See It in Action

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

U List all the main content categories we have defined on our site.

CMS Categories Overview

We currently track 4 major content pillars:

- 💻 Development (120 posts)
- 📈 Marketing (78 posts)
- 🎨 Design/UX (35 posts)
- 💡 AI & Research (62 posts)

You can narrow this down further by asking about tags or specific authors.

U Can you find all articles related to 'growth hacking'?

I found three matching content pieces that discuss growth hacking:

• **Title:** How to Growth Hack Your Codebase
Author:* Jane Doe
Published:* 2024-11-15
Tags:* #DevOps, #Strategy
Snippet:* ...focusing on rapid iteration and minimal viable product development.
• **Title:** Year 2 Recap
Author:* John Smith
Published:* 2024-12-01
Tags:* #Marketing, #Strategy
Snippet:* ...highlighting the early success of our rapid growth initiatives.

Would you like me to pull up the full text for any of these?

U What are the main content types we track, especially custom ones?

We manage several structured data sources. In addition to standard blog posts and pages, we explicitly track:

- **Global Collections:** This tracks rules for core product documentation.
- **Custom Pages:** These hold specialized models like 'Client Case Studies' or 'Technical Specs', which have unique JSON outputs.
- **Authors:** We also maintain a roster of all published authors and their associated content.

Frequently Asked Questions

01 **How does the ButterCMS MCP help me audit my website's content structure?**

The MCP gives you a single view of your entire CMS data. You can use it to list all categories, tags, and collections in one go, letting you map out how every piece of content relates without manual effort.

02 **I need to debug custom pages; is the ButterCMS MCP right for this?**

Yes. The MCP lets engineers inspect raw page models using 'list_custom_pages'. This returns explicit JSON objects, so you can pinpoint why a component isn't rendering nested data correctly.

03 **What if I only want to search for articles by a keyword?**

You simply ask your agent to perform a full-text search. The MCP runs the query and pulls back all matching blog posts, giving you immediate results based on keywords or structural properties.

04 **Can I use this MCP for SEO analysis in my content strategy?**

Absolutely. You can easily run searches to check if target keywords already exist on your site using 'search_blog_posts'. This helps you avoid duplicating content and plan backlinking strategies.

05 **Does the ButterCMS MCP handle author details, too?**

Yes. You can request a list of all published authors using the dedicated tool, which is helpful for compiling resource pages or checking authorship history across your site.

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 `"buttercms": { "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

ButterCMS 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 ButterCMS. 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	ButterCMS MCP
Server ID	`019d7566-564e-73f0-bb60-4d88aa4c5501`
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/buttercms.