MCP SERVER | NO CODE | CLOUD HOSTED

Directus MCP for AI Agents

Manage and audit structured database collections with natural conversation

Directus connects your open-source data platform and headless CMS to any AI agent. It lets you manage complex structured data—creating records, auditing schemas, and reading collections—all through natural conversation. You get full programmatic control over your PostgreSQL database without writing SQL queries.

A+ Quality Score 100/100

headless-cms | database-management | sql-interface | api-generation | asset-management | rbac

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

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.

Directus MCP

10 tools available
Cloud-hosted on Vinkius

Working with a large database often means jumping between dashboards, running complicated SQL, or asking a developer for a simple change. This MCP lets you skip that headache entirely.

Instead of digging into the Directus UI or drafting complex queries, you talk to your AI client and it handles the data layer. You can ask your agent to list all items from a specific collection, audit the structure of a table, or update a user profile by name—all in plain language. It's like giving your AI agent direct, safe access to the backend truth.

Because this MCP exposes every core function of Directus, you can manage everything from creating new content records to deleting old ones and verifying file uploads. When you plug it into Vinkius, you get a single connection point that lets any compatible client read, write, and audit your entire data platform without the usual friction points.

It's pure backend control delivered through chat.

Core Capabilities

01 — Create new CMS records

Write a new record to a collection by providing a JSON payload.

02 — Delete CMS records permanently

Irreversibly delete an existing record from the database.

03 — Check collection structure details

Inspect the properties and tables that make up a specific data collection.

04 — List all field configurations

See every detailed column configuration for any given table.

05 — Get a single item record

Retrieve the full details of one specific database row using its ID.

06 — List all available collections

See an enumeration of every structured table defined in your PostgreSQL instance.

07 — Get media file metadata

Verify and retrieve information about uploaded files stored by Directus.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP, then provide your specific Directus Base URL and Static Token.

02 Connect the credentials within your preferred AI client (Claude, Cursor, etc.).

03 Start asking questions or making requests about your data platform using natural language conversation.

The bottom line is, you're giving your AI agent a structured, conversational API layer for your existing database.

Built For

This MCP targets developers and administrators who are tired of context switching. If your job involves auditing schemas, updating CMS content, or managing users without writing boilerplate SQL, this is for you.

Database Administrator

Auditing schema definitions or managing user permissions without needing to write complex database queries.

Full-stack Developer

Creating, updating, and deleting structured records (like product data or articles) directly from an IDE or chat window.

Product Manager

Monitoring the state of collection data and verifying field configurations across different environments to plan features.

What Changes When You Connect

01 You eliminate writing complex SQL. Instead of crafting `SELECT * FROM...` queries, you simply ask your agent to find the data you need.

02 Full control over content lifecycle: Use tools like 'create_cms_record' and 'patch_cms_record' to update or modify records instantly without UI clicks.

03 Audit everything. Need to know what fields exist? Running 'list_collection_fields' gives you the schema details fast, helping development teams plan updates.

04 Manage digital assets programmatically. Use tools like 'list_directus_files' to check file uploads and generate download paths for your front-end builds.

05 Simplify user management. Easily see who has access by running 'list_directus_users', keeping track of system identities without logging into the admin panel.

06 Faster iteration cycles. Quickly validate database integrity or retrieve single items using 'get_single_item' so you can confirm data points immediately.

Real-World Applications

Content audit for a new feature

A product team needs to know what collections exist before building a new page. They ask their agent, and it runs 'list_collection_items', providing a complete list of all available tables so they can map out the data structure.

Updating bulk pricing data

The e-commerce team needs to change the price for 50 products. Instead of manually editing them, they ask their agent to use 'patch_cms_record' with a list of IDs and new prices.

Investigating data inconsistencies

A developer finds mismatched user entries. They run 'get_single_item' on the suspicious UUIDs, allowing them to compare records side-by-side in the chat interface for quick validation.

Onboarding a new admin

The operations manager needs to check which users have access. They ask their agent to run 'list_directus_users', getting an immediate, comprehensive list of authorized administrative identities.

Patterns to Avoid

Over-relying on the UI

✗ AVOID

The user logs into Directus's web interface and clicks through three different tabs (Collections > Fields > Users) just to find out how many records exist.

✓ INSTEAD

Instead, ask your agent to run 'list_collection_items' first. Then, if you need details, use 'get_collection_details'. This keeps the whole process in one conversation.

Using generic API calls

✗ AVOID

The developer writes a general payload update function that fails because it doesn't know which fields are actually available for modification.

✓ INSTEAD

First, use 'list_collection_fields' to inspect the exact schema and validate every possible column name. Then, craft your updates using 'create_cms_record' or 'patch_cms_record'.

Attempting manual data cleanup

✗ AVOID

A user tries to delete old records by manually copying IDs and writing a script that might miss required checks.

✓ INSTEAD

Let your agent handle it. Use 'wipe_cms_record' when you are absolutely certain the record must be dropped, guaranteeing irreversible deletion through a controlled command.

The Right Fit

Use this MCP if your primary need is to manage or audit structured data within an open-source CMS without writing SQL. You should connect this if you frequently perform tasks like checking schemas ('list_collection_fields'), reading specific records ('get_single_item'), or bulk updating content across collections.

Don't use it if your goal is merely to view a simple dashboard or read an article that doesn't require data manipulation. For those cases, simply connecting your AI client directly to the CMS's public API endpoint might be enough. You only need this level of deep access when you are acting as a developer or admin and must write structured payloads to maintain data integrity.

Directus MCP: Managing Structured Data in Headless CMS Platforms

Today, updating content or running reports often means logging into the Directus dashboard. You jump from the Collection View to the Field Settings, and then maybe you have to check the user directory just to get a single piece of data. It's a cycle of clicks that takes forever.

With this MCP, the process changes completely. Instead of navigating multiple screens, you simply tell your agent what needs updating—whether it's provisioning new content via 'create_cms_record' or checking if a field can handle a specific payload—and you get an immediate, actionable result.

Directus MCP: Auditing Users and Schemas for Database Integrity

Manually auditing database schemas is tedious. You have to run multiple queries just to see every single table, verify field types, and confirm which user roles are active. Mistakes here lead to bad data down the line.

This MCP lets you audit everything conversationally. Running 'list_schema_collections' gives you a clean list of all tables, while 'list_directus_users' keeps your access control lists accurate—all without writing a single `SHOW TABLES` command.

Directus: 10 Tools for Structured Database and Schema Management

Use these tools to perform every kind of data operation on your CMS, from listing all available tables to updating individual content records.

#	TOOL	DESCRIPTION
01	`create_cms_record`	Creates a new record in any Directus collection using structured JSON input.
02	`wipe_cms_record`	Permanently deletes and removes an existing database row from the specified collection.
03	`get_collection_details`	Extracts structural details, showing all properties associated with a specific data collection.
04	`list_collection_fields`	Inspects the deep internal arrays that define which fields accept formatting and validate against your database links.
05	`get_single_item`	Retrieves all explicit data for one specific row using its unique UUID identifier.
06	`list_schema_collections`	Lists every structured rule defining the PostgreSQL tables that power your CMS.
07	`list_directus_files`	Verifies and retrieves structural data about all files uploaded to the media storage.
08	`list_collection_items`	Identifies the available collections, showing which tables exist inside your CMS mappers.
09	`list_directus_users`	Lists all active user accounts and administrative identities within the database layer.
10	`patch_cms_record`	Updates an existing record in a collection by submitting new values via its unique ID.

See It in Action

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

U What are the current articles published in the blog and what fields do they use?

Articles Collection Schema

- ID: (UUID)
- Title: Text field, required.
- Author: Reference to User UUID.
- Publication Date: Timestamp.
- Body Content: Rich text editor.

I found 14 records in this collection. The most recent is 'The Future of AI' (ID: a5b8c...).

U I need to update the product 'X-700'. The new price is $199 and it's out of stock. Can you do that?

The record for 'X-700' was successfully updated.

- **Product ID:** 8c4d...
- **Status:** Out of Stock (Updated)
- **Price:** $199.00 (Updated)
- **Last Modified:** [Current Timestamp]

The database is now consistent with the new inventory data.

U Show me the structure of the user profile collection.

User Profile Collection Schema

- `user_uuid` : UUID (Primary Key)
- `first_name` : String (Required)
- `last_name` : String
- `department` : Dropdown selection.
- `avatar` : Media file reference.

This collection is used to track user details across the entire application.

Frequently Asked Questions

01 **How can I use the Directus MCP to read my CMS data?**

You can ask your agent to retrieve specific records or list all contents of a collection. It's as simple as asking, 'Show me the articles from last month,' and getting structured JSON results back immediately.

02 **Does this Directus MCP help me manage my database structure?**

Yes, it lets you audit your entire data platform. You can list all available collections or run field discovery checks to understand exactly what fields are available in a table without logging into the backend.

03 **Can I use Directus MCP to update records in bulk?**

Absolutely. You can instruct your agent to modify multiple existing rows at once, such as changing inventory counts or updating status fields across dozens of products using a single command.

04 **Is this enough for developers who write code with Directus?**

Yes. This MCP acts like an API wrapper in plain language. It allows you to test data writes, read schemas, and manage content directly from your chat or IDE, making the development loop much faster.

05 **What if I need to delete a record using Directus MCP?**

The tool handles permanent deletion for you. You just tell it which record ID needs to be vaporized, and it executes the irreversible drop safely within your agent's context.

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 `"directus": { "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

Directus 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 Directus. 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	Directus MCP
Server ID	`019d7586-6ebf-717f-9f17-e89eb9d2c6bd`
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/directus.