

MCP SERVER

NO CODE

CLOUD HOSTED

# Baserow MCP for AI Agents

Manage no-code relational databases through conversation

Baserow connects your AI agent directly to your no-code relational databases, giving you full command over structured data management. Forget manual spreadsheet entries; ask your agent to list applications, find specific records by keywords, or update project statuses across multiple tables using natural conversation.

**A+** Quality Score 100/100

no-code

relational-database

data-schema

api-integration

data-orchestration

workspace-management



# 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](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Baserow MCP

9 tools available

Cloud-hosted on Vinkius

This MCP lets you treat your entire database collection like a single conversational entity. You stop clicking through dashboards and start talking to your data. Your AI agent acts as an expert database engineer, handling complex tasks that used to take hours of manual work.

Need project statuses updated across five different tables? Just ask the agent. Want to find every record belonging to 'John Doe' while cross-referencing his payment plan and associated department? The agent runs the query for you. It manages everything from listing your high-level workspaces to updating individual fields in specific rows.

Because this MCP is hosted on Vinkius, you connect your preferred AI client once and gain access to Baserow's full suite of data management tools. You'll find that instead of digging through complex relations or struggling with limited spreadsheet views, the AI handles the structure so you can focus purely on making decisions.

---

## Core Capabilities

**01 — List all databases and workspaces**

The agent shows you a directory of every application and workspace within your Baserow account.

**03 — Modify and update structured data**

The agent handles creating new rows or updating existing fields in a table with simple instructions.

**05 — Orchestrate data flow across applications**

The agent allows you to monitor your entire Baserow ecosystem programmatically, managing everything from high-level workspaces down to individual fields.

**02 — Read specific records by criteria**

You can search tables using keywords or fetch the full details for any given row ID, no matter how deep the data is buried.

**04 — Manage database structure details**

You can request a list of all tables within a main database, or list the specific field names available in any given table.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/baserow-alternative](https://vinkius.com/mcp/baserow-alternative) — connect your AI agent in three steps.

- 01 Subscribe to this MCP on Vinkius. Then, retrieve a Database Token directly from your Baserow settings.
- 02 Connect that token and your AI client (like Cursor or Claude) through the Vinkius platform.
- 03 Ask your agent natural language questions—for example, 'Find all customers in California who haven't paid their invoice.' The agent executes the necessary reads and writes.

The bottom line is that you stop interacting with Baserow through a graphical user interface; you talk to it directly using your AI client.

---

## Built For

This MCP is for anyone who works with structured data but hates the repetitive nature of spreadsheet management. It's perfect for analysts, project managers, and developers who need an external brain to manage complex backends without leaving their workflow.

### Data Analyst

You use this MCP to automatically collect structured information from multiple related tables. Instead of writing SQL or manually exporting data, you ask the agent to pull and compile specific records into a report format.

### Project Manager

When project milestones change, you command the agent to update statuses across dozens of interconnected project boards and task trackers instantly. You track progress by asking for filtered lists rather than clicking through every view.

### Developer/Integrator

You integrate Baserow's no-code backend into custom workflows. The agent acts as the bridge, allowing your code to read and write data using simple queries instead of needing complex API calls for every operation.

## What Changes When You Connect

- 
- 01 Update records instantly: Use the `update_row` tool to change statuses or values across multiple projects without opening a single dashboard. You just tell your agent what needs changing.

---

  - 02 Eliminate manual data collection: Instead of copying and pasting information from different sources, use the agent's search capabilities ( `list_rows` ) to find specific records using natural language keywords.

---

  - 03 Understand your schema instantly: If you forget which fields are available in a table, run `list_fields` . The agent gives you an immediate inventory so you know exactly what data points you can access and manipulate.

---

  - 04 Orchestrate the entire backend: By listing applications ( `list_applications` ) and workspaces ( `list_workspaces` ), your agent acts like a system architect, giving you control over the whole database structure, not just one table.

---

  - 05 Deep dive into records: Need to confirm who is responsible for a specific task? Use `get_row` to retrieve every detail about that single entry, including all associated metadata.
- 

---

## Real-World Applications

### Finding project blockers across multiple boards

A PM needs to know which projects are stalled. The agent runs a query using `list_rows` and filters the results for 'Status: Blocked' across all relevant tables, generating an immediate list of owners and required next steps.

### Migrating old project data into new schemas

A developer needs to move records. The agent first uses `list_fields` to understand the target schema, then executes multiple `create_row` commands with clean JSON payloads for bulk entry.

### Updating client status after a meeting

An analyst attends a kickoff call and needs to update records. They ask the agent to find all clients associated with 'Acme Corp' and use `update_row` to set their status to 'Follow-up Scheduled'.

### Auditing user permissions and structure

An ops engineer needs a security check. They ask the agent to list all available applications (`list_applications`) and retrieve metadata about database tokens, verifying who has write access.

---

## Patterns to Avoid

---

### Treating data like isolated spreadsheets

#### ✗ AVOID

The user manually requests the agent to list rows from Table A, then asks it separately to list rows from Table B. The results are two separate lists that require manual comparison.

#### ✓ INSTEAD

Instead of listing them in sequence, ask the agent: 'Find all records where the project ID in Table A matches a client ID in Table B.' This leverages the relational context automatically.

### Forgetting what fields exist

#### ✗ AVOID

The user tries to update a row by specifying a field name (e.g., 'Budget Cap') that doesn't actually exist, resulting in an API error and lost time.

#### ✓ INSTEAD

Before updating any record, ask the agent to run `list_fields` on the target table first. This guarantees you have the correct, current list of available field names.

### Over-relying on simple searches

#### ✗ AVOID

The user only runs a search query (`list_rows`) using basic keywords and gets 500 results, none of which are useful without further filtering or context.

#### ✓ INSTEAD

Use the agent to first run `list_tables` to narrow down the scope. Then ask: 'In the 'Client Data' table, list rows where the status is open AND the last contact date was within the last 30 days.' Be specific about both criteria.

---

## The Right Fit

Use this MCP if your workflow involves managing structured data across multiple interconnected tables and you want to avoid repetitive UI clicks. You need an agent that can act as a database query engine, capable of listing applications, finding records using keywords (`list_rows`), and modifying data via `update_row`. Don't use it if all you need is a simple contact list—a general file

management tool will suffice. If your main pain point is just viewing *one* single table, then a dedicated spreadsheet alternative might be enough. But when the problem involves orchestrating relationships between different project components or workspaces, this MCP is essential.

---

## Baserow MCP: Managing No-Code Database Schemas with AI Agents

Today, managing structured data means clicking through multiple tabs, cross-referencing different sheets, and manually updating records across separate project trackers. You spend more time figuring out *where* the information is than actually using it.

With this MCP, you simply tell your agent what needs to change—for example, 'Update all pending client leads to Stage 2.' The agent handles the complex query logic, finding and updating every single row across your interconnected tables instantly. You get back a confirmed list of changes.

---

## Baserow MCP: Automating Data Retrieval from Complex Workspaces

The manual process requires you to remember which application holds the 'contact info' and which one holds the 'payment history,' forcing you into a constant cycle of opening, navigating, and comparing data sources.

Now, your agent treats all these applications as one unit. You ask for combined context—like retrieving records based on criteria across multiple tables in one go. The AI delivers the comprehensive view without any manual switching or copy-pasting.

---

# Baserow: 9 Tools for Data Schema Management

These tools allow your AI agent to list applications, read specific rows, or modify complex relational data within your databases.

#	TOOL	DESCRIPTION
01	<code>list_applications</code>	Retrieves a list of all primary databases (applications) set up in your Baserow account.
02	<code>create_row</code>	Adds an entirely new record to a specific table, requiring you to provide the field name and its value in JSON format.
03	<code>delete_row</code>	Permanently removes a specified row from any given table.
04	<code>list_fields</code>	Provides an inventory of all available field names and types within a selected table.
05	<code>get_row</code>	Fetches the complete, detailed data for one single row using its unique ID.
06	<code>list_rows</code>	Searches and lists multiple rows in a table, supporting pagination and keyword filtering.
07	<code>list_tables</code>	Shows all the individual tables that exist inside one main database application.
08	<code>update_row</code>	Modifies existing data in a specific row by providing new values for selected fields.
09	<code>list_workspaces</code>	Displays an overview of all major working areas and project groupings within your account.

---

# See It in Action

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

**U** Show me all active databases and workspaces I have set up.



## Your Baserow Ecosystem Overview

I found 3 primary applications (databases) and 1 main workspace:

- **Main Workspace:** Client Management Hub
- \* **Applications:**
  - \* **Customer CRM** : Tracks client details, history.
  - \* **Project Tracker** : Manages milestones and tasks.
  - \* **Inventory List** : Holds product SKUs and stock levels.
- **Database Tokens Available:** 3 (Active)

You can ask me to look at any of these for specific tables. Which one should we start with?

**U** Find all project records that are currently marked 'At Risk' in the Project Tracker.



## Search Results: At-Risk Projects

I found 5 projects flagged as 'At Risk.' Here are the details:

ID	Project Name	Client Contact	Reason for Risk	Last Update
P102	Alpha Launch	Jane Doe	Funding delay (Q3)	2024-06-01
P215	Beta Rollout	John Smith	Scope creep detected	2024-05-28

Would you like me to draft an internal memo for the Project Manager regarding these items?

**U** Can you update Jane Doe's status and add a note that we rescheduled her call?



### Update Confirmation

✓ Success. I updated row ID `CRM-102` in the 'Customer CRM' database.

**Field Updated:** Status set to: Follow-up Scheduled\*

- **Timestamp:** 2024-06-12
- **Note Added:** "Rescheduled call with Jane Doe. New date confirmed for next week."

The record is now live.

---

## Frequently Asked Questions

---

### 01 How do I find my Baserow Database Token?

Log in to your account, navigate to **Settings** > **Database Tokens**, and create a new token with appropriate workspace permissions.

### 02 Can I search for records via AI?

Yes! The `search_rows` tool allows your agent to find records across a specific table matching your search criteria programmatically.

### 03 How do I find Table and Database IDs?

Use the `list_applications` tool to find Database IDs, and `list_database_tables` to find Table IDs within a specific application.







---

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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
 <b>Claude AI</b>	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
 <b>Cursor</b>	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
 <b>VS Code</b>	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add <code>"baserow-alternative": {   "url": "..." }</code>
 <b>Windsurf</b>	MCP Settings → <code>mcp_settings.json</code> → Add endpoint URL
 <b>ChatGPT</b>	Settings → Tools & plugins → Add MCP server → Paste endpoint
 <b>Gemini</b>	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

# Baserow 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](https://vinkius.com) · [support@vinkius.com](mailto: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 Baserow. 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	Baserow MCP
Server ID	019dd0c0-7b6e-7396-82db-286f4d96a8ae
Platform	Vinkius Cloud for AI Agents
Endpoint	<a href="https://edge.vinkius.com/{token}/mcp">https://edge.vinkius.com/{token}/mcp</a>

### 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/baserow-alternative](https://vinkius.com/mcp/baserow-alternative).