

MCP SERVER

NO CODE

CLOUD HOSTED

Coding.net MCP for AI Agents

Manage Code Repositories, Issues, and Projects via Conversation

Coding.net connects your AI agent to a major DevOps platform, letting you manage complex software development lifecycles through natural conversation. Your agent can instantly list projects, audit Git repositories, track bugs, and monitor merge request status without needing to navigate the web interface.

A+ Quality Score 100/100

git-repository

issue-tracking

software-lifecycle

collaboration

devops-platform

code-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 — connect your AI agent in under 60 seconds.

Coding.net MCP

10 tools available

Cloud-hosted on Vinkius

This MCP lets your AI client treat core DevOps tasks like chatting with a teammate. You don't have to click through dashboards or remember complex API calls; you just ask for what you need. Your agent handles connecting to projects, diving into Git metadata, and keeping tabs on issue progress, all in one conversation. Whether you're reviewing code history or coordinating a release across multiple teams, your AI assistant acts as a real-time DevOps expert. It consolidates project orchestration, repository auditing, and issue tracking into something simple enough for natural language interaction. Getting access to this capability via the Vinkius catalog means you connect once and get full visibility into your entire software development workflow.

Core Capabilities

01 — Discover all DevOps projects

List every organizational unit in the platform and retrieve detailed metadata for each project.

03 — Track bugs and features

Get a complete listing of project issues or retrieve granular details on a specific task or bug report.

05 — Understand user roles

Retrieve profile data for organization members and the current authenticated user.

02 — Audit Git code repositories

Browse existing Git depots, list branches, and get detailed information about specific commits.

04 — Monitor code reviews

List active merge requests to quickly identify pending reviews, roadblocks, or required changes.

One Click on Vinkius — From Prompt to Execution

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

- 01 Subscribe to this MCP in Vinkius.
- 02 Enter your CODING Team Domain and Personal Access Token into your AI client's settings.
- 03 Ask your agent a question—for example, 'What are the latest issues for Project Alpha?'—and it executes the workflow.

The bottom line is that you manage your entire DevOps lifecycle by talking to your AI client, not by clicking through interfaces.

Built For

This MCP is for engineers and leads who hate context switching. If you spend half your day navigating multiple internal dashboards just to answer a simple status question, this connector saves you time. It lets you keep all your code history, project data, and bug reports in one conversation.

DevOps Engineer

Automates repository audits and monitors CI/CD pipelines by asking the agent to check commit details or list branches.

Project Manager

Tracks issue progress, lists projects, and checks team velocity without leaving their chat window.

Engineering Lead

Coordinates code reviews by listing active merge requests and retrieving detailed repository metadata across multiple teams.

What Changes When You Connect

- 01 Instead of opening the web UI to get a list of projects, you ask your agent to list all DevOps projects directly. This saves multiple clicks.

-
- 02 You can audit code history instantly. Asking for commit details lets you know exactly who changed what and when, without navigating Git logs.

 - 03 Monitoring team velocity is faster. You simply request a list of issues or check project metadata to see how many tasks are open.

 - 04 Code review coordination gets easier. Your agent lists active merge requests, immediately identifying blockers or stalled pull requests.

 - 05 Getting basic user data is simple. Use the `get_user` tool to retrieve profile information for any member in your organization.
-

Real-World Applications

Checking project status before a meeting

The PM needs to know if the 'Alpha' backend is ready. The agent lists projects, gets repo details for 'main-api', and checks if any active merge requests are stuck waiting on review.

Onboarding a new team member

The lead needs to show the newcomer the project structure. The agent lists all DevOps projects, retrieving metadata for each one so the lead can give an immediate overview of the scope.

Investigating a production bug

The engineer finds a critical bug reported in the system. They use `get_issue` to gather all details, then use `list_branches` and `get_commit` to trace which specific code commit caused the issue.

Coordinating a hotfix release

A critical patch is needed. The engineer uses `list_mrs` to see what's currently merging, then uses `get_repo` and `list_branches` to select the correct stable branch for deployment.

Patterns to Avoid

Deep-diving into logs manually

X AVOID

Spending 15 minutes navigating through Git's web interface, clicking commit by commit just to find out who changed the API endpoint last week.

✓ INSTEAD

Use the `'get_commit'` tool. Ask your agent directly for 'the person who committed changes to the API module between June 1st and June 3rd'. It pulls the data instantly.

Forgetting which project has the bug

X AVOID

A PM sees a ticket but can't remember if it belongs to 'Mobile App' or 'E-commerce Backend', wasting time searching multiple dashboards.

✓ INSTEAD

Start by asking your agent to `'list_projects'`. Once you see the list, you can immediately narrow down where that bug originated.

Mixing up repository details

X AVOID

Copying metadata from one repo (like 'main-api') and pasting it into a different project's documentation by hand.

✓ INSTEAD

Use `'get_repo'` for the correct project. The agent retrieves all required metadata in a clean, structured format you can trust.

The Right Fit

You should use this MCP if your team relies heavily on managing code history, tracking bugs, or coordinating multiple development teams across different repositories. If your workflow requires checking issue progress (`get_issue`), auditing branches (`list_branches`), or listing merge requests (`list_mrs`) frequently, this is for you. Don't use it if you only need to manage simple user directories; other tools might be better suited. However, if your primary need is managing complex project relationships—like knowing which issues belong to which project unit and tracking the code behind them—this MCP connects all those pieces into one conversation.

Managing DevOps Code Repositories with Coding.net

Today, checking on a development project means jumping between five different tabs: the main dashboard for issues, the repository viewer for code details, the branch selector, and several separate merge request pages. You copy-paste IDs and manually cross-reference dates to build a single status report.

With this MCP, you just ask your agent, 'Give me an overview of Project Alpha's development readiness.' It aggregates information from the `list_projects`, `get_repo`, and `list_mrs` tools. You get a unified, accurate summary without opening a single web tab.

Tracking Software Development Issues with Coding.net

Writing status reports means manually pulling ticket numbers from the issue tracker and then going to Git to check if the corresponding code commit exists, often resulting in outdated or incomplete data.

Now, you can ask your agent to 'List all open bugs for Project Beta'. It runs `list_issues` and checks relevant commits using `get_commit`. You get a real-time report that links the reported bug directly to its development history. That's accurate status reporting.

10 Tools for DevOps Project and Issue Tracking

Use these tools to list projects, audit commits, check branches, track bugs, and monitor all aspects of your code repository status.

#	TOOL	DESCRIPTION
01	<code>get_commit</code>	Fetches the specific details, including authors and dates, of any given commit.
02	<code>get_issue</code>	Retrieves comprehensive information for a single project issue or bug report.
03	<code>get_project</code>	Retrieves specific details for a single DevOps project using its identifier.
04	<code>get_repo</code>	Gets detailed metadata about a specified code repository.
05	<code>get_user</code>	Fetches the profile details, including roles and names, for the currently authenticated user.
06	<code>list_branches</code>	Retrieves all current branches for a chosen repository.
07	<code>list_issues</code>	Lists multiple project issues, providing an overview of open tasks and bugs.
08	<code>list_mrs</code>	Provides a list of active merge requests to track code integration status.
09	<code>list_projects</code>	Lists all available DevOps projects within the organization.
10	<code>list_repos</code>	Lists every repository associated with a given project.

See It in Action

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

U List all my DevOps projects on CODING.



✓ Project Overview

You currently manage 5 active organizational units:

- E-commerce Backend : Primary domain for online sales.
- Mobile App : Handles iOS and Android frontends.
- Infrastructure : Core services (DB, auth).
- Analytics Engine : Data processing unit.
- Marketing Site : Static web pages.

Which project do you want to drill down into?

U Show me the latest issues for project 'Alpha'.



🚨 Project Alpha Issue Tracker

Here are 12 open tasks. High Priority items require immediate attention:

ID	TYPE	TITLE	STATUS
#453	Bug	Login Timeout on Mobile	Open
#460	Feature	Dark Mode Implementation	Needs Review
#471	Bug	Payment Gateway Failure	Critical

Would you like details for the critical failure (#471)?

U Get the branches for repository 'main-api' in project 'Alpha'.



Repository Branches: main-api

Three active branches are available:

- `master` : The production-ready code base.
- `develop` : Current development work, unstable.
- `feature/api-v2` : Work for the next major version (WIP).

I found 3 recent commits across these. Do you want to check the commit history on `master` ?

Frequently Asked Questions

01 How can I use the Coding.net MCP to check project status?

Your agent lets you list all projects and retrieve metadata instantly. You can ask for an overview of a specific unit, getting details like its current owners or last activity without leaving your chat.

02 Can I use Coding.net MCP to track down old code changes?

Yes. By using the commit tools, you can ask for specific commit details—who made a change, when they did it, and what files were affected—all based on natural language queries.

03 What if I need to know which feature is blocked in my development cycle?

You can list active merge requests. This immediately surfaces pending reviews or required changes that are acting as blockers for the team's progress.

04 Does Coding.net MCP help with bug tracking and issue resolution?

It does. You can get a complete listing of all project issues, retrieve details on specific bugs, and track their status to monitor how quickly the team is resolving them.

05 Is Coding.net MCP just for code, or does it handle user data too?







It handles both. Beyond repositories and commits, you can retrieve profile information and general metadata about organization members using this MCP.

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 <code>"codingnet": { "url": "..." }</code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → 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

Coding.net 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 · 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 Coding.net. 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	Coding.net MCP
Server ID	019d8427-b84a-7393-a60f-b55cffefcc3c
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/codingnet.