MCP SERVER | NO CODE | CLOUD HOSTED

PingCode MCP

Manage sprints, projects, and docs conversationally.

PingCode MCP connects your AI agent directly to an agile project management platform for R&D teams. It lets you manage work items, track sprints, monitor releases, and retrieve documentation without opening a dashboard. You talk naturally, and it updates your entire development pipeline.

A+ Quality Score 100/100

agile | sprint-planning | issue-tracking | rd-management | wiki | workflow-automation

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

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.

PingCode MCP

10 tools available
Cloud-hosted on Vinkius

Forget clicking through endless dashboards just to find out what's happening with a feature or who owns a bug. This MCP connects your AI agent to PingCode, the platform R&D teams use for complex project coordination. Instead of juggling tabs to list projects, check sprints, and then search the wiki, you talk to your agent like talking to a team lead who already knows everything. Your agent handles all that complexity: it lists active agile projects, creates new tasks with full descriptions, checks sprint progress, or pulls specific documentation from the knowledge base. It makes the entire software development lifecycle feel less like navigating complicated forms and more like having a natural conversation. Vinkius hosts this MCP so you can connect PingCode to any compatible client you already use. You get an R&D assistant that keeps your pipeline moving and your documentation instantly accessible.

Core Capabilities

01 — Track Projects and Work Items

List all available agile projects and create new work items, such as tasks or bugs, directly through chat.

02 — Monitor Development Cycles

View active sprints, track upcoming releases, and get details about the overall project scope.

03 — Manage Team Structure

List all organizational teams and members to quickly figure out who is assigned to what.

04 — Access Project Documentation

Find and retrieve specific content from the wiki or list all available knowledge repositories.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP, then enter your PingCode Client ID and Client Secret.

02 Connect your preferred AI client (Claude, Cursor, etc.) using the credentials you just set up.

03 Tell your agent what you need; for example, 'List all active sprints for the Mobile App V3 project.'

The bottom line is, once connected, your AI agent handles the API calls and data parsing so you don't have to.

Built For

The Product Owner who needs a status report without opening five different dashboards. The Developer who can track down documentation or create a bug ticket mid-flow. The Agile Coach overseeing multiple team backlogs from one single interface.

Product Owner

You use this to audit the backlog, check sprint velocity, and get an instant overview of which projects are ready for release.

Software Developer

You rely on it to pull up technical specs from the wiki or file a bug report without leaving your coding environment.

Agile Coach

You use this to oversee multiple team workflows, listing members and comparing sprint statuses across several departments simultaneously.

What Changes When You Connect

01 Stop switching contexts. Instead of opening PingCode to list agile projects and then switching over to the wiki to find specs, your agent handles both requests instantly in one chat window.

02 Never lose track of a bug again. You can ask your agent to create_work_item for a new issue, automatically capturing all necessary metadata and assigning it immediately.

03 Get an instant project health check. With list_sprints and list_releases, you get a clear view of the delivery schedule without running reports or clicking through status dashboards.

04 Knowledge is easy to find. Instead of guessing which wiki page holds the answer, use list_wiki_pages and then get_wiki_page to pull exact documentation content on demand.

05 Understand ownership immediately. By using list_teams and list_members, you can figure out who owns a piece of code or who is assigned to a task without manual lookups.

Real-World Applications

A new feature requires immediate documentation.

The developer asks the agent: 'What are the auth requirements for V3?' The agent uses list_wiki_pages and then get_wiki_page to pull the correct section from the System Architecture page, saving a half-hour of manual searching.

A high-priority bug needs tracking.

The Product Owner simply types: 'Create a critical bug for payment failure in the checkout flow.' The agent executes create_work_item and automatically sets the priority, ensuring the right people see it instantly.

Need to update team assignments quickly.

The Agile Coach asks: 'Who is on the Core Engine Refactor team?' The agent runs list_teams and then lists_members to give a comprehensive roster, helping reassign tasks faster than manual email chains.

Checking overall product readiness.

A manager asks: 'What's the status of the next two releases?' The agent uses list_releases and combines it with list_sprints to give a full, aggregated view of the delivery pipeline.

Patterns to Avoid

Treating the MCP like a search engine.

✗ AVOID

Trying to copy-paste entire project status reports into the chat window and asking the agent to 'process this.'

✓ INSTEAD

Don't paste. Ask direct questions instead, such as: 'List all work items in the Checkout Flow project,' or 'Get project details for Core Engine Refactor.'

Overlooking the necessary setup.

✗ AVOID

Assuming the agent knows which PingCode organization you're working on without providing credentials first.

✓ INSTEAD

Always connect your AI client using the provided Client ID and Secret. This authenticates the connection so the agent can access the correct data via list_projects.

Confusing tools with documentation.

✗ AVOID

Asking for a project status update, but only getting raw, unformatted text from an old wiki page that doesn't reflect current sprints.

✓ INSTEAD

Use the specific workflow functions first. Ask 'List sprints,' and if you need background context, then ask to get_wiki_page.

The Right Fit

You should use this MCP if your daily job involves moving between multiple complex views in PingCode: project boards, documentation wikis, task lists, and team directories. If you are constantly switching tabs or copy-pasting status updates into Slack, this tool saves time by centralizing the data access layer. You need it when your workflow is defined by *status checking* or *creation*. Don't use it if you simply want to write a blog post about agile methodologies; for that, general knowledge models are fine. If your main goal is only communicating with an external ticketing system like Jira, then a dedicated issue tracking MCP would be better. But if the problem is coordinating development and documentation across *multiple* domains within PingCode, this is what you need.

The Pain of Context Switching

Today's workflow requires jumping between five different views: the project list to see scope, the board to track tasks, the wiki to find specs, and a separate team directory just to know who to ping. You end up copying status updates from one screen and pasting them into another, losing crucial context or getting bogged down in manual searching.

With this MCP, you talk to your agent once. Instead of clicking through screens, you simply ask: 'Show me the bugs for Project Alpha,' then follow up with, 'And what is the wiki page on API endpoints?' You get all that project data and documentation instantly, without ever leaving the chat.

PingCode MCP Gives You Full R&D Visibility

The tedious manual steps disappear. No more running separate reports to list projects; no need to manually check for upcoming releases by comparing sprint dates with the overall project timeline.

Now, your development pipeline is managed through natural conversation. You stop managing buttons and start talking to your system.

PingCode with 10 Tools

Use these tools to perform specific actions like creating tasks or retrieving project data directly from your AI client.

#	TOOL	DESCRIPTION
01	`create_work_item`	Creates a new task, story, or bug item in the project management system.
02	`get_project`	Fetches detailed information about a specific agile development project.
03	`get_wiki_page`	Retrieves the full content of a specified wiki page by its name or ID.
04	`list_members`	Provides a list of all people belonging to the organization and their roles.
05	`list_projects`	Retrieves names and details for every agile project currently set up in PingCode.
06	`list_releases`	Lists all planned or completed releases associated with a specific project.
07	`list_sprints`	Shows the active and upcoming sprints for an agile development project.
08	`list_teams`	Lists all departmental teams within the organization structure.
09	`list_wiki_pages`	Provides a searchable list of available wiki pages across the entire platform.
10	`list_work_items`	Retrieves a paginated list of all work items, like bugs or tasks, within a defined project scope.

See It in Action

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

U List all agile projects in my PingCode organization.

I've retrieved your PingCode projects. You have 5 active agile projects, including 'Core Engine Refactor' and 'Mobile App V3'. Which one would you like to view items for?

U Create a new bug item in project 'Checkout Flow' titled 'Payment timeout on mobile'.

Done! I've created a new bug item 'Payment timeout on mobile' in the 'Checkout Flow' project. I've also set the item type to 'bug' for you. Would you like to add a description?

U Retrieve the content of the wiki page 'System Architecture' from repository 'PROJ-DOCS'.

I've retrieved the 'System Architecture' page. It contains detailed diagrams and component descriptions for the current architecture. Would you like me to summarize the key modules?

Frequently Asked Questions

01 **How do I list all my available projects using PingCode MCP?**

You use the list_projects tool. This gives you a comprehensive view of every agile project defined in your organization, helping you narrow down where to focus your efforts.

02 **Can PingCode MCP create tasks or bugs? What is the correct tool?**

Yes, you can use the create_work_item tool. This allows you to define a new piece of work—a bug, story, or task—and push it directly into your project backlog.

03 **Does PingCode MCP know where my technical documentation is?**

Yes, the list_wiki_pages tool lets you browse available knowledge areas. After finding the correct page, use get_wiki_page to pull the actual content.

04 **I need to check team assignments; which tool should I use with PingCode MCP?**

You can list_teams first to see all departments, and then list_members will provide a roster of people belonging to those teams or projects.

05 **What if I want to know the status of sprints and releases together? Can PingCode MCP handle that?**

You can combine list_sprints with list_releases. This allows your agent to give you a timeline view, showing what development cycles are active versus when the features ship.

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 `"pingcode": { "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

PingCode 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 PingCode. 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	PingCode MCP
Server ID	`019d846c-2873-72ee-b504-baf935a4c2ff`
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/pingcode.