MCP SERVER | NO CODE | CLOUD HOSTED

Gitea MCP

Control project code, issues, and branches via chat.

Gitea MCP connects your self-hosted Git instance directly to any AI agent. You can manage repositories, track issues, and audit pull requests—all from conversation. Quickly list projects, check organization details, or verify branch protection rules without leaving your chat window.

A+ Quality Score 100/100

git | version-control | self-hosted | code-hosting | issue-tracking

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

SCAN TO VIEW

The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect 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.

Gitea MCP

10 tools available
Cloud-hosted on Vinkius

Connect your private Gitea instance using this MCP, and you'll get full control over project collaboration right through any AI client. Instead of logging into the web interface, you talk to your agent and ask it to perform tasks like checking repository details or listing all active issues in an organization. You can track everything from pull request status to user profile information using natural language.

It lets your agent list every accessible repository with its clone URL and visibility state. Need to audit team progress? Your agent pulls up a comprehensive list of issues across multiple repos, letting you see labels, states, and assignees at a glance. This powerful connection is managed through the Vinkius catalog, giving you access to Gitea's full suite of developer tools without ever having to click away from your IDE or chat window.

Core Capabilities

01 — View all project repositories

List every repository you have access to and get detailed info like the clone URL, star count, and if it's private or public.

02 — Track issues and tasks

See a list of all open issues in any repo, checking numbers, labels, states, and who is assigned to them. You can also pull the full text body for deep analysis.

03 — Audit code review requests

List every pull request—open, closed, or merged—and check which branches they connect from and to.

04 — Manage organizational structure

See all organizations you belong to, pulling metadata like the organization's website and total repository count.

05 — Check branch rules and status

List every branch in a repo and verify commit SHAs and whether specific protection rules are active on that branch.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP, then provide your Gitea Instance URL and Access Token (you find the token in your Gitea User Settings > Applications).

02 Your agent establishes a secure connection, allowing it read access to your code data.

03 You talk to your AI client—it uses the connection to pull repository lists, issue statuses, or organization details on demand.

The bottom line is you get Git project oversight without needing to open a browser tab.

Built For

This MCP is for anyone who spends time managing codebases and tracking work across multiple projects. If your workflow involves checking PR status, listing issues in different teams, or auditing branch permissions, this saves you context switching.

Developer

Needs to check the open pull requests for a specific feature branch and list all related development issues without leaving their IDE.

DevOps Engineer

Requires verification of branch protection rules across multiple repositories and needs to confirm the full details of an organization's structure.

Product Manager

Uses it to audit project issues, compiling reports on how many tasks are stuck in 'Review' status across three different teams.

What Changes When You Connect

01 You stop context switching. Instead of opening the web UI to check repository details or listing all repos, you just ask your agent, and it pulls out clone URLs and visibility status immediately.

02 Issue tracking becomes instant. Asking for a list of issues lets you see numbers, states, labels, and assignees across multiple projects without clicking through dashboards.

03 Code review auditing is simplified. Your agent can list all pull requests, letting you monitor if they're open or merged and verifying the source/target branches instantly.

04 Organization visibility improves. Use this MCP to identify all organizations you belong to and get high-level metadata about them, like their location and total repo count.

05 Compliance checks are easier. You can list every branch in a repository and check its protection rules or verify commit SHAs just by asking your AI client.

Real-World Applications

Auditing an entire feature set

A developer needs to confirm that all related components are ready for merge. They ask the agent to list pull requests for 'auth-service', then use a tool to get full details on #45, and finally check the branch protection rules using `list_branches`. The entire audit happens in one chat thread.

Onboarding new team members

A manager needs an overview of all projects. They ask the agent to list organizations (`list_orgs`), then select 'Marketing', and finally request a list of all repositories belonging to that org using `list_org_repos`. This gives them instant scope visibility.

Diagnosing stale tasks

A product manager notices delays. They ask the agent to list issues across 'Eng-Team' and then filter those results by label or assignee, immediately pinpointing where work is stuck.

Patterns to Avoid

Copying URLs manually

✗ AVOID

A user opens the Gitea web UI to find a repo's clone URL and then has to copy it into an external planning document.

✓ INSTEAD

Ask your agent to use `list_repos`. It returns all necessary details, including the full clone URL, directly in the chat output for immediate pasting.

Checking branch status one by one

✗ AVOID

A DevOps engineer needs to check if three specific branches have required protection rules enabled across two different repos.

✓ INSTEAD

Use `list_branches` and then ask the agent to verify the effective branch protection rules for all of them in a single query.

The Right Fit

Use this MCP if your core workflow revolves around Git project management: listing, auditing, or tracking. Specifically, if you need to check PR status (`list_pulls`), view organization membership (`list_orgs`), or see granular branch details (`list_branches`). Don't use it if your goal is just basic code viewing—you still need a standard SSH client for that. If you only manage project documentation without tracking issues, an issue-tracking tool focused on tickets might be better suited instead of this Git-specific MCP.

The hassle of leaving the chat to check code status

Right now, if you need to know the status of a feature branch or list all issues for a project, your workflow hits a wall. You have to open up a new browser tab, navigate to the repository dashboard, click into 'Issues,' and then manually search through tabs until you find what you were looking for.

With this MCP, that manual process disappears. You simply ask your agent: 'What are the active issues in the frontend repo?' It pulls the list of issues right into your chat window—no clicks needed.

Gitea MCP delivers instant visibility into project data

You no longer have to juggle tabs just to get a basic overview. Listing repositories using `list_repos` or confirming an organization's membership via `list_orgs` is now a single, conversational command.

It moves project oversight from tedious clicking into actionable conversation. You keep your focus on the work, not on the UI.

Gitea: 10 Tools for Code Management

Use these tools within your agent to perform specific actions like listing repositories or getting detailed information on individual issues.

#	TOOL	DESCRIPTION
01	`list_repos`	Lists every repository you can see, providing names, descriptions, clone URLs, star counts, and visibility status for each one.
02	`get_repo`	Retrieves the complete set of details for a single specified Gitea repository.
03	`list_issues`	Generates a list of all current issues within a specific repository, including their status, labels, and who is assigned them.
04	`get_issue`	Fetches the full content and detailed metadata for one specified issue number.
05	`list_orgs`	Provides a list of all organizations you are currently a member of within Gitea.
06	`get_org`	Retrieves detailed information about a specific organization, including its website and total repo count.
07	`list_org_repos`	Lists every repository that belongs to a specified Gitea organization.
08	`list_pulls`	Gathers all pull requests for a given repo, letting you check if they are open, closed, or merged.
09	`list_branches`	Lists every branch in a repository and provides associated commit SHAs to confirm its history.
10	`get_me`	Extracts your own authenticated Gitea user profile identity, including your email and administrative status.

See It in Action

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

U List the last 5 repositories I worked on

Retrieving your repositories... I found 5 recent projects including 'web-app', 'api-service', and 'infra-config'. Highlights: 'web-app' (Default: main, Visibility: private). Which one would you like to see the issues for?

U Show me open pull requests for repo 'api-service'

Retrieving PRs for 'api-service'... I found 2 open pull requests: #101 'Add OAuth support' and #102 'Fix rate-limiting bug'. Would you like the review status or branch details for either of these?

U List all issues in organization 'Eng-Team'

Retrieving organization issues... I found 15 issues across the 'Eng-Team' repositories. Most are in 'core-engine' and 'frontend-v2'. Would you like me to filter them by labels or assignees?

Frequently Asked Questions

01 **How do I use Gitea MCP to check if a repo is private?**

Use `list_repos`. The tool returns full details for every repository you can see, including a status flag that shows whether it's marked as 'private' or 'public'.

02 **Can I track issues in an organization using Gitea MCP?**

Yes. First, use `list_orgs` to find the correct organization ID, and then use tools like `list_org_repos` followed by `list_issues` to gather all relevant task data.

03 **Does Gitea MCP work for reading PR details?**

Absolutely. The `list_pulls` tool gathers all open, closed, or merged pull requests in a repository so you can audit the review status.

04 **What is the best way to verify branch rules with Gitea MCP?**

Use `list_branches`. This function lists every branch and allows you to verify commit SHAs and check if specific branch protection rules are in place for auditing purposes.

05 **Is the user profile visible via Gitea MCP?**

Yes. You can run the `get_me` tool, which extracts your authenticated profile identity, giving you access to your login name, email, and admin status.

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 `"gitea": { "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

Gitea 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 Gitea. 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	Gitea MCP
Server ID	`019d75a5-3df3-7214-8813-106fa3c0e64a`
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/gitea.