MCP SERVER | NO CODE | CLOUD HOSTED

Markdown Task Extractor MCP

Aggregate tasks from your entire local vault.

Markdown Task Extractor scans local directories, pulling every open and completed to-do item from any Markdown note. Whether you use Obsidian, Logseq, or Notion exports, this MCP treats your entire file vault as one centralized task list. It processes hundreds of files in milliseconds, giving your AI client a complete picture of what needs doing across all your projects.

A+ Quality Score 100/100

task-tracking | markdown | glob-pattern | todo-list | note-taking | task-aggregation

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/markdown-task-extractor

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.

Markdown Task Extractor MCP

1 tools available
Cloud-hosted on Vinkius

Your notes aren't organized in one place; they're scattered across dozens of daily journals and project folders. If you ask an agent to find your pending tasks without this MCP, it fails because it can't read those local files. This connector solves that problem by scanning a specified folder path. It pulls every task marked with `- [ ]` or `- [x]` from all the Markdown notes within, regardless of how old or where they sit on your computer. The result is a clean, structured list that you feed directly into your AI chat context. With Vinkius managing this MCP in their catalog, you connect once and gain access to powerful file processing like this. You stop copying task lists from five different files just to ask one question about your overall workload.

Core Capabilities

01 — Aggregate Tasks From Folders

Scans a specified directory path, pulling all open and completed tasks marked in Markdown notes.

02 — Identify Task Statuses

Distinguishes between pending items (open) and completed items using standard markdown syntax.

03 — Handle Large Volumes of Files

Uses fast file globbing patterns to read thousands of local Markdown files quickly, providing high-speed context feeding.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/markdown-task-extractor** — connect your AI agent in three steps.

01 Provide the MCP with the absolute directory path containing all your raw Markdown notes.

02 The system scans every file in that folder, extracting only lines that match the standard task syntax (`- [ ]` or `- [x]`).

03 Your AI client receives a consolidated text block: one complete list of tasks from across your entire local vault.

The bottom line is you get a single, structured data dump containing every to-do item from all the Markdown files in your specified folder.

Built For

Knowledge workers and researchers who live in local markdown vaults. If your idea of organization involves copying and pasting tasks from multiple daily notes, you need this. It targets people whose workflow is constantly interrupted by scattered information.

Product Manager

Needs to survey dozens of project files to build a comprehensive list of required features or outstanding blockers.

Technical Writer

Uses this MCP to pull every action item from their research notes across multiple drafts and guides before writing the final document.

Researcher/Student

Gathers all reading tasks, follow-up ideas, and bibliography actions from scattered lecture notes or article summaries.

What Changes When You Connect

01 Stop hunting through files. This MCP pulls every task, open or closed, into one place so your agent can analyze your full workload instantly.

02 It doesn't just read text; it understands Markdown syntax. It correctly differentiates between pending tasks (`- [ ]`) and finished items (`- [x]`).

03 The speed is insane. Using fast glob patterns, you process hundreds of files in milliseconds—you won't wait for your agent to load the context.

04 You don't have to manually list file paths or worry about syntax. Just point the MCP at your notes folder and get a clean task dump.

05 It works with standard note-taking systems, including Obsidian vaults, Notion exports, and Logseq files.

Real-World Applications

Project Retrospective

A PM needs to know what was supposed to happen on a feature launch that spanned three months. They ask their agent to scan the entire project folder, and the MCP provides a list of all pending tasks marked up over time, allowing them to pinpoint missed steps.

Weekly Planning

A technical writer has finished drafting several guides into different folders. They use the MCP to consolidate every single action item and follow-up thought from the entire directory so they can plan their next week's writing schedule.

Deep Research Review

A student has collected articles and notes across multiple local folders for a thesis. Instead of manually tracking progress, they use the MCP to gather all research tasks into one list and ask their agent which ones need immediate attention.

Bug Tracking Review

A developer needs to review every reported issue across several sprint folders. The MCP pulls out all action items, letting them quickly see the status of every bug fix without reading hundreds of comments.

Patterns to Avoid

Using basic text search

✗ AVOID

Manually searching for 'task' or using a general file reader that just dumps all the raw content, making it impossible to separate action items from meeting notes.

✓ INSTEAD

Use `extract_markdown_todos`. This tool specifically scans for Markdown task syntax (`- [ ]` and `- [x]`) and isolates only the actionable items, keeping your context clean.

Copying across apps

✗ AVOID

Opening five different notes (one from Notion, one from Obsidian, etc.) and manually copying all the tasks into a single spreadsheet or chat window.

✓ INSTEAD

Connect this MCP to your AI client. You provide the folder path once, and the tool aggregates everything for you automatically.

Relying on cloud sync

✗ AVOID

Assuming that simply having files in a synced cloud folder means an agent can access them without specific configuration or permission.

✓ INSTEAD

This MCP is built to read local file paths. Ensure the source data lives locally, and provide the absolute directory path to `extract_markdown_todos`.

The Right Fit

Use this if your core problem is aggregating scattered action items from a large collection of Markdown files stored on your local machine. You need an agent to read hundreds of notes and identify tasks based on specific markdown syntax (like `- [ ]`). Don't use it if you are working with highly structured data in a database or a cloud-only service that doesn't support file path scanning; for those, look into specialized API connectors. Also, don't expect it to understand *context* beyond the task marker—it will pull everything marked as a task, whether relevant or not. It is a powerful extraction layer, designed solely to feed structured text context to your existing AI client.

Your local knowledge base is everywhere, and nothing is organized.

Right now, finding what you need feels like digging through an archaeological dig site. You open Obsidian for project notes, switch to your Notion exports for meeting minutes, then check Logseq for daily thoughts. To answer a simple question like 'What's my biggest priority?' you have to manually copy and paste every single task from every relevant file into one place just so the AI can read it.

With this MCP, that process vanishes. You give your agent the folder path once. The tool scans hundreds of Markdown files in milliseconds, pulling out only the tasks—open or closed—and delivering a clean, unified list directly to your chat context.

Markdown Task Extractor: Get One Single List of Everything

The most time-consuming part of note-taking is the inventory. You spend hours opening file after file, trying to track down every incomplete thought or action item that was written weeks ago.

Now you can simply ask your agent a question about your workload. The MCP handles the entire aggregation process, giving you immediate clarity without any manual data wrangling.

Markdown Task Extractor: 1 Tool Available

Use the tools here to scan folders and aggregate all to-do items from Markdown files into a structured list for your agent.

#	TOOL	DESCRIPTION
01	`extract_markdown_todos`	Scans a local directory path to extract every open and completed task marked in Markdown files, listing the file source for context.

See It in Action

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

U Scan my Obsidian vault at C:/Notes and list all my pending tasks grouped by file.

I found 12 pending tasks across 4 different files. Here is the grouped breakdown:

U Look through my Notion exports folder and tell me how many tasks I completed this week.

Based on the `- [x]` markers, you have completed exactly 24 tasks.

U Find all tasks in my project folder that contain the hashtag '#urgent'.

I found 3 open tasks marked with #urgent:
1. Fix database deployment (backend.md)
2. Call investor (meetings.md)

Frequently Asked Questions

01 **Can Markdown Task Extractor read tasks from PDF files?**

No, this MCP only scans Markdown (`.md`) files. It relies on the specific markdown syntax used in those text files; it cannot process non-text documents like PDFs.

02 **What file types does extract_markdown_todos support?**

It supports any local directory containing Markdown files, including those exported from apps like Obsidian, Logseq, and Notion.

03 **How fast is this MCP?**

It uses advanced glob patterns to scan large directories extremely quickly, processing hundreds of files in milliseconds. You don't have to wait long for context loading.

04 **Does Markdown Task Extractor handle completed tasks?**

Yes, it distinguishes between open and closed items. It extracts both pending (`- [ ]`) and completed (`- [x]`) tasks, giving you a full history.

05 **Do I need to write custom code to use this MCP?**

No. You just connect your preferred AI client to the Vinkius catalog and tell it the local folder path. The agent handles the rest.

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 `"markdown-task-extractor": { "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

Markdown Task Extractor 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

V I N K I U S

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Markdown Task Extractor. 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	Markdown Task Extractor MCP
Server ID	`019e38bc-eea0-701a-858d-23e195c267da`
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/markdown-task-extractor.