

M C P S E R V E R

N O C O D E

C L O U D H O S T E D

Library Fine Calculator MCP

Automate overdue fee calculations and status checks.

Library Fine Calculator MCP handles the math and logic behind overdue book penalties. Your AI client uses this to determine exact fine amounts, check if items are still in grace periods, and provide detailed breakdowns of charges. It's a precise way to manage library penalties without manual calculations.

A+ Quality Score 100/100

library

finest

overdue

calculation

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.

Library Fine Calculator MCP

4 tools available

Cloud-hosted on Vinkius

You can hand off the tedious math of penalty management to your AI client. This MCP gives your agent the ability to process overdue items by calculating specific fine amounts based on days late. Instead of manually checking grace periods or calculating daily accruals, you just provide the item details and the fee structure. Your agent can then tell you if an item is still within a grace period, if it has hit a maximum penalty cap, or exactly how much is owed. It also includes a way to validate that your fee and period settings make sense for a library system, ensuring your logic stays consistent. Whether you are managing a small collection or a large database, this tool keeps your penalty tracking accurate and fast.

Core Capabilities

01 — Fine Calculation

Your agent calculates the exact dollar amount owed for overdue items.

02 — Grace Period Detection

The AI checks if an item is still within its allowed grace period before charging fees.

03 — Penalty Caps

Your client identifies when an item has reached the maximum possible fine.

04 — Logic Validation

The tool verifies that fee and period parameters are consistent with library rules.

05 — Detailed Summaries

Your agent generates breakdowns of chargeable days and total amounts.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/en/ai-agent-connect/library-fine-calculator — connect your AI agent in three steps.

- 01 Connect the MCP to Claude, Cursor, or Windsurf via Vinkius.
- 02 Ask your AI client to calculate a fine or check a status.
- 03 The agent invokes the necessary tool from the MCP.
- 04 The tool returns the calculation or status data.
- 05 Your agent presents the final result to you.

Connect the MCP to your AI client to start running fine calculations immediately.

Built For

This MCP is built for library administrators and staff who need to automate penalty tracking.

Librarians

Hand off fine calculations and status checks to your AI agent.

Library System Developers

Use the validation tool to test fee and period logic.

Collection Managers

Let your agent summarize overdue statuses across multiple items.

What Changes When You Connect

- 01 Eliminates manual math for overdue fine calculations.
- 02 Automates the detection of grace periods and penalty caps.

-
- | | |
|----|---|
| 03 | Provides instant breakdowns of chargeable days and total amounts. |
| 04 | Validates fee parameters to ensure logical consistency. |
-

Real-World Applications

Automated Fine Calculation

Your agent calculates the exact fine for a patron based on their specific overdue duration.

Status Monitoring

Check if a specific book is currently in a grace period or accruing daily fees.

System Testing

Verify that your library's fee and period settings are logically valid.

Patron Reporting

Generate a detailed summary of fines for a patron to show exactly how their total was calculated.

Patterns to Avoid

Manual Math Guesswork

❌ AVOID

You try to estimate a patron's total debt by looking at the due date and multiplying it by a daily rate in your head.

✓ INSTEAD

Let your agent do the math using `get_fine_amount` to ensure the calculation is exact and accounts for all rules.

Ignoring Grace Periods

❌ AVOID

You immediately charge a user a fine the moment an item is one minute past its due date.

✓ INSTEAD

Use `check_overdue_status` to see if the item is still within its allowed grace period before applying any fees.

Unverified Fee Logic

❌ AVOID

You set a new daily fine rate in your database without checking if it conflicts with your maximum penalty cap.

✓ INSTEAD

Run `validate_fine_parameters` to confirm your fee and period settings are logically consistent with library rules.

The Right Fit

Library administrators and system developers should connect this MCP to automate penalty tracking. If you are a casual reader or a user looking to pay a single fine, you don't need this. The decision depends on whether you need to manage, validate, or automate fine logic across a collection.

Library Fine Calculator 4 Tools

These tools handle the math and logic for overdue items, grace periods, and penalty caps.

#	TOOL	DESCRIPTION
01	get_fine_amount	Calculates the total fine owed for a specific item based on how many days it is overdue.
02	get_fine_summary	Provides a detailed breakdown of the overdue status, including the days subject to charging and the final amount.
03	validate_fine_parameters	Verifies that the provided fee and period parameters are logically consistent and valid for a library system.
04	check_overdue_status	Determines if an item is currently within a grace period, accruing fines, or has hit the maximum penalty.

See It in Action

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

U How much is the fine for an item 10 days overdue with a \$2 daily fee and a 3-day grace period?



The total fine is \$14.00.

U What is the status of an item that is 2 days late with a 5-day grace period?



The item is within the grace period.

U Give me a summary for 15 days overdue, \$5 daily fee, 2-day grace period, and a \$40 cap.



Chargeable days: 13, Total fine: \$40.00, At cap: true.

Frequently Asked Questions

01 How does this MCP calculate fines?

It uses the number of overdue days, the daily fee, and any grace periods to determine the total amount owed.

02 Can it handle maximum fine limits?

Yes, the tool can account for a maximum penalty cap and will report if an item has hit that limit.

03 Which AI clients can use this MCP?

You can use this with any MCP-compatible client, including Claude, Cursor, and Windsurf.

04 Does it support grace periods?

Yes, it can determine if an item is within a grace period and whether it is currently accruing fines.

05 Can I use it to test my library's fee settings?

Yes, the `validate_fine_parameters` tool checks if your fee and period settings are logically consistent.

06 How does the fine calculation work?

The fine is calculated by multiplying the days overdue that exceed the grace period by the daily fee, up to the specified maximum fine cap.

07 Can I set a maximum limit for fines?

Yes, you can use the `maxFineCap` parameter in tools like `get_fine_amount` to ensure the penalty does not exceed a certain amount.

08 What is a grace period?

A grace period is a set number of days after the due date during which no fines are accrued.

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>"library-fine-calculator": { "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

Library Fine Calculator 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 Library Fine Calculator. 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	September 2026
MCP Server	Library Fine Calculator MCP
Server ID	01a0c560-2a94-736b-b606-6d19dc9b5e55
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/en/ai-agent-connect/library-fine-calculator.