MCP SERVER | NO CODE | CLOUD HOSTED

Everhour Time Tracking MCP

Audit budgets, monitor team hours in conversation.

Everhour Time Tracking lets your AI agent manage projects, track team time, and monitor budgets using natural conversation. Get instant summaries of project finances, see who's logged hours, and audit utilization rates without opening the web app.

A+ Quality Score 100/100

time-tracking | budget-monitoring | team-productivity | billing-status | task-management

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/everhour-time-tracking

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.

Everhour Time Tracking MCP

10 tools available
Cloud-hosted on Vinkius

This MCP connects Everhour, a powerful time tracking system, directly to your AI client. You can run audits on entire teams or specific projects by just asking questions. Instead of navigating multiple dashboards, you prompt your agent for details like total hours logged this week or which projects are running low on budget. For example, you can get a list of every active project and its current status, or monitor real-time timers across your organization. When you connect Everhour through Vinkius, your AI client handles all the heavy lifting, turning complex time sheets into simple answers in your chat window.

Core Capabilities

01 — Audit team productivity

Get a high-level summary of recent time entries and check which projects are currently active.

02 — Check project budgets

Review detailed settings, budget limits, and current utilization rates for any specific project.

03 — Track live work sessions

See exactly what task is running right now and when the timer started.

04 — Review team time records

List all recorded hours for your entire team within a specific date range.

05 — Identify projects at risk

Get a clear list of only those projects that are still within their assigned budget or time limits.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/everhour-time-tracking** — connect your AI agent in three steps.

01 Connect the Everhour integration to your AI client and authorize it using your API key.

02 Tell your agent what you need—for example, 'Show me all projects over budget' or 'List team hours for last month'.

03 The MCP runs the necessary checks against the Everhour data and delivers a readable summary directly into your chat.

The bottom line is that you get instant, actionable insights about your time and money without ever leaving your messaging platform.

Built For

Project Managers who are constantly checking budget burn rates. Agency Owners needing to reconcile client billing hours quickly. Operations Leads overseeing team capacity.

Project Manager

Uses the MCP to check project budgets and total team time entries on the fly, ensuring projects stay profitable.

Agency Owner

Requests reports detailing client billing status and accumulated hours across multiple accounts via chat.

Operations Director

Monitors organizational team capacity, checking running timers and overall productivity metrics instantly to manage workload distribution.

What Changes When You Connect

01 Stop digging through dashboards. Asking your agent to list all projects gives you instant access to budget status and details for every single one.

02 Understand where time is actually going. You can request a summary of recent entries or check which tasks are running right now using get_currently_running_timer.

03 Never miss a billing opportunity. The MCP lets you list all clients configured for invoicing, keeping your accounts receivable clean and current.

04 Manage team workload easily. You can pull comprehensive reports listing time records for the whole team across defined dates, helping spot underutilized staff.

05 Keep track of project health automatically. Use list_projects_within_budget to get a quick roster of projects that are safe from overspending.

Real-World Applications

Determining client billing status

An agency owner needs to know the total hours logged for a specific retainer client. Instead of running three reports and exporting them, they prompt their agent: 'What are the total billable hours for Client X this month?'. The MCP runs through list_billing_clients and aggregates the time data instantly.

Checking team bandwidth before starting work

A project manager is assigned a new, large task. They ask their agent to check 'What are our available hours for Q3?'. The MCP uses list_organization_team_members and list_team_time_records to calculate the true capacity remaining.

Reviewing project scope creep

The client claims a project is going over budget. The PM asks their agent: 'Show me all projects exceeding 90% of their budget.' The MCP uses list_projects_within_budget and get_project_detailed_data to pinpoint the exact problem areas.

Auditing daily activity

An operations lead needs a quick snapshot of team output. They prompt: 'Give me today's productivity summary.' The MCP uses quick_time_tracking_audit, giving an immediate picture of task completion and active work.

Patterns to Avoid

Forgetting the right tool

✗ AVOID

A user tries to find out if a project is over budget by just asking 'Is anything too expensive?'. The agent doesn't know where to start because the request is too vague.

✓ INSTEAD

To check budgets, use get_project_detailed_data for specific projects or list_projects_within_budget to see all safe and unsafe options.

Mixing up team data with project data

✗ AVOID

A user asks the agent about a 'team's budget,' leading to confusion because budgets are tied to specific projects, not groups of people.

✓ INSTEAD

To get team hours, use list_team_time_records. To check spending limits, always reference a specific project using get_project_detailed_data.

Ignoring the current status

✗ AVOID

The user forgets that work is happening right now and asks for historical data instead of real-time updates.

✓ INSTEAD

If you need to know what's happening *right now*, use get_currently_running_timer. It instantly gives the task name and start time.

The Right Fit

Use this MCP if your core problem is correlating who spent how much time on which project, against a set budget. You need to move beyond simple reporting and actually run audits—like identifying all projects that are underutilized or those running dangerously close to their spending limit. Don't use it if you simply need to create new tasks; this MCP only reads data. If your goal is just to list every client name without any financial context, then a basic CRM integration might suffice. But if you need to cross-reference team member roles (using list_organization_team_members) with their historical time records and compare that against project budgets (get_project_detailed_data), this is the right tool.

Tracking Time and Budgets Used to Be a Spreadsheet Nightmare

Today, checking your team's budget status means logging into Everhour, navigating through multiple project folders, pulling up financial reports, and manually cross-referencing those numbers against the actual hours logged. You spend fifteen minutes just gathering data that should take five seconds.

With this MCP, you simply ask your agent to list all projects currently over budget. The system handles the multi-step process of checking utilization rates across all accounts and delivers a clear answer: two projects are running too hot. You get answers, not reports.

Everhour Time Tracking Gives You Real-Time Project Oversight

Manually auditing team time requires pulling separate logs for each person and date range (list_team_time_records). Then, you have to manually compare the total against the allocated budget limit found in get_project_detailed_data. It's a tedious cycle of copy-pasting.

Now, your agent runs that entire audit instantly. You ask for 'all active projects and their status,' and it combines all that data into one comprehensive update. That’s how you save time.

Everhour Time Tracking: 10 Tools for Project Management

These tools let you pull specific data points—from individual project details to full team time records—so your AI agent can build complete financial and productivity reports.

#	TOOL	DESCRIPTION
01	`get_everhour_user_metadata`	Pulls your personal profile and metadata information from Everhour.
02	`get_project_detailed_data`	Retrieves the complete budget and setup details for one specific project.
03	`quick_time_tracking_audit`	Provides a quick summary of recent time entries and active projects right away.
04	`get_currently_running_timer`	Shows the task name and start time for any timer that is currently running.
05	`list_billing_clients`	Lists all clients set up in Everhour for billing and invoicing purposes.
06	`list_tracked_projects`	Generates a list of every project currently tracked within your account.
07	`list_project_tasks`	Retrieves all tasks associated with a specific, defined project.
08	`list_organization_team_members`	Lists every team member and their assigned role in the organization.
09	`list_team_time_records`	Gathers time records for a group of people within a specified date range.
10	`list_projects_within_budget`	Identifies and lists all projects that are currently operating under their assigned budget or time limit.

See It in Action

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

U List all projects currently over budget.

I've identified 2 projects exceeding their budget: 'Website Redesign' (115% utilization) and 'Q1 Marketing Campaign' (102% utilization). Would you like to see the breakdown of hours by team member for the Website Redesign?

U Show me the tasks for project 'Mobile App'.

The 'Mobile App' project has 12 tasks, including 'User Authentication' (8h logged), 'API Integration' (15h logged), and 'UI/UX Design'. Total time logged for this project is 45 hours. Should I pull the estimated vs. actual time for these tasks?

U What is the team productivity summary for this week?

This week, the team has logged a total of 120 hours across 5 active projects. 'John Doe' has logged the most hours (35h), followed by 'Jane Smith' (32h). Average daily productivity is 6 hours per member. Would you like a list of any missing time entries?

Frequently Asked Questions

01 **How do I check if a project is over budget with Everhour Time Tracking?**

You can ask the agent to list projects exceeding their budget, or use get_project_detailed_data for one specific project. This gives you the percentage utilization immediately.

02 **Does Everhour Time Tracking help me monitor who is working right now?**

Yes, use get_currently_running_timer. It instantly shows the task name and start time for any active timer across your organization.

03 **Can I list all my clients in Everhour Time Tracking?**

Absolutely. The list_billing_clients tool gathers every client configured for billing, helping you audit accounts receivable quickly.

04 **How do I get a team's total hours for last month?**

Use the list_team_time_records tool. You just specify the desired date range and the group of people, and it compiles all recorded time automatically.

05 **What is the easiest way to see project status?**

The quick_time_tracking_audit tool provides a high-level summary instantly. It's perfect for getting a general feel for recent activity and current projects.

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 `"everhour-time-tracking": { "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

Everhour Time Tracking 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 Everhour Time Tracking. 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	Everhour Time Tracking MCP
Server ID	`019d7593-aa13-7278-b0e3-04145eef73b1`
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/everhour-time-tracking.