MCP SERVER | NO CODE | CLOUD HOSTED

Garment Cost Per Wear Tracker MCP

See the actual value of every item in your closet.

Garment Cost Per Wear Tracker MCP lets your AI client track how often you wear your clothes and how much each use actually costs you. Instead of looking at the sticker price, you can see which pieces are true investments and which ones are just taking up space. It handles the math and the history so you can focus on building a more efficient wardrobe.

A+ Quality Score 100/100

fashion | budgeting | wardrobe | cost-per-wear | efficiency

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · SEPTEMBER 2026
vinkius.com/en/ai-agent-connect/garment-cost-per-wear-tracker

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.

Garment Cost Per Wear Tracker MCP

4 tools available
Cloud-hosted on Vinkius

Stop guessing if that expensive jacket was worth it. This MCP turns your AI client into a personal wardrobe economist by tracking the relationship between what you paid and how often you actually use your clothes. You can log every time you pull an item out of your closet and let your agent do the heavy lifting on the math. It calculates the cost per wear for specific items, compares different pieces to see which offers better long-term value, and provides a high-level view of your entire collection's utility. It's a simple way to change how you think about shopping and clothing ownership. By keeping a running tally of usage, you move past the initial purchase price and start seeing the real economic impact of your fashion choices.

Core Capabilities

01 — Usage Tracking

Your agent logs every time you wear an item to keep your data current.

02 — Investment Comparison

The AI compares two items to determine which one provides more utility for the price.

03 — Wardrobe Auditing

Your agent summarizes the efficiency of your whole closet at once.

04 — Unit Cost Math

The MCP calculates the exact cost of a single use for any specific garment.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/en/ai-agent-connect/garment-cost-per-wear-tracker** — connect your AI agent in three steps.

01 Connect the MCP to your preferred client like Claude or Cursor via Vinkius.

02 Tell your agent about a new clothing purchase and its price.

03 Log wear events as you use your clothes.

04 Ask your agent to compare items or summarize your total wardrobe efficiency.

Connecting this MCP to your AI client gives you an instant way to manage wardrobe economics.

Built For

This is for anyone looking to make more intentional clothing purchases and manage a budget through usage data.

Budget-conscious shoppers

They hand off usage logs to the AI to see if their purchases align with their spending goals.

Capsule wardrobe enthusiasts

They use the agent to ensure every item in their minimal collection is being utilized effectively.

Sustainable fashion advocates

They track wear frequency to reduce waste and prioritize high-quality, long-lasting pieces.

What Changes When You Connect

01 Identifies high-value items through usage data.

02 Reduces impulse buys by focusing on long-term cost.

03 Provides a clear overview of wardrobe utility.

04 Automates the math behind clothing depreciation.

Real-World Applications

Pre-purchase Research

Compare a potential new purchase against your current best-value items to see if it's a smart buy.

Daily Logging

Tell your agent you wore a specific shirt today to update its lifetime cost.

Closet Audits

Ask for a summary of your wardrobe to find items that have a high cost per wear and aren't being used.

Budget Planning

Use the efficiency data to decide where to allocate your next clothing budget.

Patterns to Avoid

Manual Math Guessing

AVOID

You try to estimate if a jacket was worth it by doing mental math every time you look at it in the mirror.

✓ INSTEAD

Let the MCP do the math for you. Use `calculate_cpw` to get the exact cost per wear based on your actual usage history.

Vague Wardrobe Audits

AVOID

You look at your closet and decide you have too many clothes without knowing which ones actually provide value.

✓ INSTEAD

Get a clear picture of your utility. Use `summarize_wardrobe_efficiency` to see how well you are actually using your entire collection.

Ignoring Item Comparison

AVOID

You buy a new shirt because it looks good, ignoring the fact that your current shirts are barely used.

✓ INSTEAD

Check the math before you buy. Use `compare_garment_value` to see if a new item is a better investment than what you already own.

The Right Fit

Connect this MCP if you want to stop overspending on clothes and start treating your wardrobe as a financial asset. Do not connect it if you don't care about the long-term cost of your purchases or if you prefer impulse shopping over data-driven decisions.

Garment Cost Per Wear Tracker Tools

These tools handle the math and data entry for your wardrobe. They turn simple usage logs into actionable financial data for your clothing collection.

#	TOOL	DESCRIPTION
01	`compare_garment_value`	This tool compares the cost per wear of two different items. Use it to decide which piece represents a better financial investment based on usage.
02	`summarize_wardrobe_efficiency`	This tool provides a bird's-eye view of your entire collection. It aggregates data to show how efficiently you are using your total wardrobe.
03	`track_wear_event`	This tool records a single instance of wearing a garment. It automatically updates the cost per wear for that item in your history.
04	`calculate_cpw`	This tool computes the cost per wear for a specific piece of clothing. It divides the total price by the number of times you've worn it.

See It in Action

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

U What is the cost per wear for a jacket that cost $120 and has been worn 30 times?

The cost per wear for the jacket is $4.00.

U Which is a better value: a $50 shirt worn 10 times, or a $100 shirt worn 40 times?

The $100 shirt is the better value with a cost per wear of $2.50, compared to $5.00 for the shirt.

U I just wore my $200 boots for the 5th time. What is my new cost per wear?

The new cost per wear for your boots is $40.00.

Frequently Asked Questions

01 **How does this MCP calculate cost per wear?**

It takes the total price of the garment and divides it by the number of times you have recorded wearing it.

02 **Can I use this with Claude or Cursor?**

Yes, you can connect this MCP to any compatible client including Claude, Cursor, and Windsurf.

03 **Do I need to manually do the math?**

No, your AI client handles all the calculations once you provide the price and usage data.

04 **How do I update how many times I've worn something?**

You simply tell your agent that you wore an item, and it uses the track_wear_event tool to update the history.

05 **Is this useful for capsule wardrobes?**

Yes, it helps you see which items in a capsule wardrobe are being used most frequently and which ones are underperforming.

06 **How is Cost Per Wear calculated?**

It is calculated by dividing the total purchase price of a garment by the number of times it has been worn.

07 **Can I compare two different items?**

Yes, you can use the `compare_garment_value` tool to determine which item provides better value based on its usage.

08 **How do I update my usage count?**

Use the `track_wear_event` tool to record a new instance of wearing an item, which automatically updates its CPW.

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 `"garment-cost-per-wear-tracker": { "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

Garment Cost Per Wear Tracker 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 Garment Cost Per Wear Tracker. 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	Garment Cost Per Wear Tracker MCP
Server ID	`01a0c5ad-9f2f-73bc-8889-5e072af365b6`
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/garment-cost-per-wear-tracker.