

MCP SERVER

NO CODE

CLOUD HOSTED

Gelato MCP

Manage POD Orders and Global Fulfillment Tracking

Gelato manages your entire print-on-demand (POD) workflow directly through natural conversation. This MCP lets you list active orders, fetch detailed product catalogs, track real-time shipments globally, and calculate shipping quotes before placing a new order. Manage customs paperwork, verify API connections, and keep your custom merchandise production running without ever touching the dashboard.

A+ Quality Score 100/100

print-on-demand

fulfillment

dropshipping

inventory-management

global-shipping

custom-products



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.

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.

Gelato MCP

12 tools available

Cloud-hosted on Vinkius

Managing print-on-demand fulfillment usually means hopping between five different tabs: the order list, the product catalog, the tracking page, the pricing sheet, and the webhook logs. This MCP changes that. It connects to your Gelato account so you can handle every step of the custom merchandise lifecycle using just your agent's chat window. Need to know if an item is in stock or what a specific design costs to ship? Just ask. You can pull up current product catalogs and retrieve specifications instantly, then check the real-time shipping status for any order. It also lets you place new orders and even cancel them if something changes. Because this whole setup runs through Vinkius, your agent gets instant access to all these functions, letting you manage everything from initial quote requests to final shipment tracking without ever leaving your chat client.

Core Capabilities

01 — Reviewing Account Status

You can verify the API connection and check account details to ensure continuous production flow.

03 — Product Research

You retrieve product metadata and access available catalogs to see what items you can sell.

05 — System Monitoring

You can list all configured webhooks and check account identity to keep your internal systems up-to-date with production changes.

02 — Managing Orders

The agent can list existing orders, create new print jobs, or cancel an order if necessary.

04 — Shipping & Pricing

The MCP requests real-time quotes for shipping costs across different regions, or tracks the shipment details for any given order.

One Click on Vinkius — From Prompt to Execution

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

- 01 Subscribe to this MCP through the Vinkius Marketplace.
- 02 Enter your Gelato API Key into the connection settings (found in Developers > API keys).
- 03 Start chatting with your agent; it now has access to all your POD fulfillment tools.

The bottom line is you talk to your AI client, and it executes the complex logistics tasks across Gelato for you.

Built For

This MCP is essential for e-commerce operations managers who spend too much time clicking through fulfillment dashboards. It's for support staff needing rapid tracking lookups, and creative directors who need immediate access to product specs before a client meeting.

E-commerce Operations Manager

They use the MCP to list all print orders and check their status quickly, ensuring nothing gets stuck in fulfillment.

Customer Support Specialist

They retrieve tracking numbers and order details instantly when a customer asks for an update, without having to log into the external portal.

Creative Director

They check product catalogs and fetch detailed specifications immediately to confirm if a design can be printed on a specific item type.

What Changes When You Connect

- 01 You instantly check order status by calling `get_order_status`, eliminating the need to navigate complex, multi-step dashboard menus just to see if an item is 'In Production'.

-
- 02** Instead of manually checking product guides, you use `list_product_catalogs` and then `list_catalog_products` to get a comprehensive overview of every available merchandise type.
-
- 03** When pricing a new custom run, you don't guess the cost. You call `get_shipping_quote` to instantly calculate real-time production and shipping rates for any region.
-
- 04** You keep your system accurate by using `list_print_webhooks`, ensuring that internal systems receive automated alerts whenever Gelato updates a job.
-
- 05** If an order needs adjusting, you can quickly call `create_print_order` or `cancel_print_order` right from the chat interface instead of logging into the dedicated fulfillment dashboard.
-

Real-World Applications

The Support Team Needs Tracking Data

A customer emails asking about their shipment. Instead of telling the customer to log in and check, you ask your agent to call `get_order_shipments` using the order ID. Your agent responds instantly with the carrier name, tracking number, and ETA.

Handling Bulk Order Changes

The marketing team needs to change or stop a pending large run. They ask the agent to list all active jobs with `list_print_orders`, identify the target job, and then call `cancel_print_order` immediately.

Launching a New Product Line

A creative director needs to confirm if their new design works on both hoodies and shirts globally. They use `list_product_catalogs` first, then check specific item data using `get_product_details`, confirming availability before committing to an order.

Pre-Sale Pricing Check

A client asks for a quote on 100 custom mugs to be shipped from Mexico. You use your agent to execute `get_shipping_quote`, providing the necessary region and quantity data to get an accurate, real-time price.

Patterns to Avoid

Treating it like a simple order book

✗ AVOID

Thinking you can just list orders and stop. You miss the critical tracking info or product details needed for client communication.

✓ INSTEAD

Always use ``list_print_orders`` to get the IDs, but follow up with ``get_order_shipments`` to provide actual delivery data to your user.

Guessing fulfillment costs

✗ AVOID

Manually consulting a spreadsheet for shipping rates when a new region is involved. This leads to delays or incorrect quotes.

✓ INSTEAD

When in doubt, call ``get_shipping_quote`` with the target location and product details. It provides real-time pricing across different regions.

Ignoring system updates

✗ AVOID

Assuming that an order status update was received because it happened on Gelato's side, but your internal systems are outdated.

✓ INSTEAD

Use ``list_print_webhooks`` to confirm which webhooks are configured and actively receiving production data, ensuring your entire workflow is covered.

The Right Fit

Use this MCP if your bottleneck is the sheer volume of manual data retrieval related to physical goods—specifically product catalogs, order statuses, or shipping logistics. If you need to check a specific item's availability, verify an API connection, or calculate real-time shipping costs before placing a new job, this is exactly what you need. However, don't use it if your primary goal is accounting (you need a dedicated finance MCP) or managing raw inventory counts that aren't tied to a fulfillment order; in those cases, look for an inventory management tool instead of relying on POD status reports.

The Old Way: Jumping Through Fulfillment Hoops

Today, managing custom merchandise is clicking through five separate portals. You check the order list to see if a job exists; then you switch tabs to find out what product type it is; next, you navigate to tracking to see when it left the warehouse; finally, you might open a calculator just for the shipping quote. It's constant context switching and copy-pasting.

With this MCP connected via Vinkius, that entire sequence disappears. You ask your agent one question—for instance, 'What is the status of order 456?'—and it executes `get_order_status` and `get_order_shipments` simultaneously. It gives you a single, conversational answer.

Gelato MCP: Fulfillment Data at Your Fingertips

The manual steps that vanish include logging into the internal order dashboard just to see if an item has been printed; having to cross-reference a product UID against three different pricing sheets; and manually verifying your API keys after every system update.

Now, you simply ask. The agent handles all the complex data retrieval behind the scenes, giving you immediate answers on everything from `get_product_details` to placing a new job with `create_print_order`. It's that simple.

Gelato MCP: 12 Tools for POD Operations

These twelve tools allow you to manage every phase of your print-on-demand lifecycle, from creating a new order to tracking the final shipment.

#	TOOL	DESCRIPTION
01	<code>verify_api_connection</code>	Checks your API access credentials to make sure the connection is live.
02	<code>create_print_order</code>	Places a brand new custom merchandise order with Gelato.
03	<code>cancel_print_order</code>	Cancels an existing print order if the customer changes their mind or the design is flawed.
04	<code>get_account_info</code>	Retrieves basic details about your Gelato account identity and setup.
05	<code>get_order_status</code>	Gets detailed information on a specific order's current fulfillment status.
06	<code>get_product_details</code>	Fetches comprehensive metadata for an individual product using its unique ID.
07	<code>get_shipping_quote</code>	Calculates the estimated cost of shipping and production for a potential order location.
08	<code>list_product_catalogs</code>	Shows all available product catalogs that Gelato supports, like T-shirts or mugs.
09	<code>list_print_orders</code>	Lists every print order currently recorded in your account.
10	<code>list_catalog_products</code>	Displays all the specific products available within a chosen catalog.
11	<code>get_order_shipments</code>	Tracks the real-time shipment details and carrier information for any order.
12	<code>list_print_webhooks</code>	Lists all webhooks you've configured to receive production updates from Gelato.

See It in Action

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

U List all my active print orders and their current status.



Retrieving orders... I found 5 active print orders, including 'ORD-99' (Status: In Production) and 'ORD-101' (Status: Shipped). Would you like the tracking details for the shipped one?

U Get the shipping details for order 'gelato_12345'.



Fetching shipments... Order gelato_12345 has been shipped via DHL. The tracking number is '1Z999...' and the estimated delivery is October 15th.

U List all products in the 'Apparel' catalog (ID: cat_987).



Retrieving products... In the Apparel catalog, I found 12 items including 'Classic T-Shirt' (UID: shirt_001) and 'Premium Hoodie' (UID: hood_05).

Frequently Asked Questions

01 How do I check if my API key is working with the Gelato MCP?

Call the `verify_api_connection` tool. This instantly checks your credentials and confirms that the connection between your agent and Gelato is active.

02 Can the Gelato MCP help me list all my past orders?

Yes, you can use `list_print_orders` to retrieve a complete list of every job recorded in your account. This is useful for auditing or finding old IDs.

03 What tool do I use if I need to know the shipping cost?

Use the ``get_shipping_quote`` tool. You provide the product type and destination, and it returns a real-time quote for both production and shipping.

04 Does Gelato MCP only track shipped items?

No, you can use ``list_product_catalogs`` first to see what's available. Then, if needed, you can use ``get_order_status`` to check the job status before it ships.

05 How do I add a new product type using Gelato MCP?







The MCP doesn't create products; it reads them. Use ``list_product_catalogs`` and then ``list_catalog_products`` to see every option available for you to use.

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>"gelato": { "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

Gelato 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](https://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 Gelato. 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	Gelato MCP
Server ID	019d75a3-035c-70f8-bf99-161d77150baa
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/gelato.