MCP SERVER | NO CODE | CLOUD HOSTED

PedidosYa MCP

Manage order flow and courier logistics instantly.

PedidosYa MCP automates your entire restaurant operation across Latin America's leading food delivery platform. Manage incoming orders, adjust menus on the fly, and dispatch couriers—all through natural conversation with any AI client.

A+ Quality Score 100/100

food-delivery | restaurant-operations | menu-management | logistics | order-processing

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/pedidosya

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.

PedidosYa MCP

14 tools available
Cloud-hosted on Vinkius

Managing a popular restaurant means juggling more than just cooking; you're running logistics, marketing, and inventory simultaneously. This MCP lets you treat your PedidosYa partner account like an extension of your mind. You can receive new orders and instantly accept or reject them without ever touching the physical tablet. Need to mark items as sold out because you ran low? Do it in real-time on your live menu listing. It goes deeper than just order status: you can even request a courier for internal B2B deliveries and track their exact GPS location until they arrive at your kitchen door.

If you're looking for reliable, multi-platform control over high-volume delivery operations in Latin America, Vinkius hosts this MCP so you connect once to manage everything. You can even set up automated webhooks that fire when cancellations happen or new orders come in, keeping your systems always updated.

Core Capabilities

01 — Manage Order Lifecycle

Receive and act on incoming orders by accepting them for preparation, rejecting them with a reason, or marking them as ready for pickup.

02 — Control Menu Availability

Instantly update pricing or toggle products on or off so customers only see what you actually have in stock right now.

03 — Coordinate Courier Logistics

Schedule on-demand couriers for package pickups and track their real-time location until the delivery is complete.

04 — Monitor Restaurant Health

View operational details, including all registered restaurant locations and their current business hours.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP using your PedidosYa Partner Client ID and Secret.

02 Connect the credentials to your AI client (like Claude or Cursor).

03 Ask your agent to perform an action, such as listing all pending orders or requesting a courier.

The bottom line is you use natural language prompts to execute complex operational tasks that normally require logging into a separate portal and clicking through multiple screens.

Built For

This MCP is essential for restaurant owners, dark kitchen operators, and logistics managers running multi-site or high-volume operations. If your current process requires manual tablet interaction to handle basic flow (like accepting orders or adjusting stock), you need this.

Restaurant Owner

Uses the MCP to accept incoming customer orders and mark items sold out when they run low, all without physically touching the Point of Sale tablet.

Dark Kitchen Operator

Orchestrates multi-brand virtual kitchen operations by programmatically managing menu updates across various physical venues.

Logistics Manager

Requests on-demand courier pickups for B2B deliveries and monitors the entire shipment journey in real time using this MCP.

What Changes When You Connect

01 Accept and manage orders using the `list_orders` tool. Instead of clicking through a tablet interface, you simply ask your agent to process all pending deliveries in natural language.

02 You maintain full menu control by using `update_product_status`. If an item sells out, mark it unavailable immediately so customers don't place orders for things you don't have.

03 Handle last-mile logistics with the `request_courier` tool. Schedule pickups and track shipments in real time using `track_shipment`, keeping your inventory flowing smoothly.

04 Keep your data pipelines running by setting up webhooks via `create_webhook`. This ensures that when a cancellation happens, your internal systems are instantly notified without needing manual checks.

05 Adjust pricing on the fly. The `update_product_price` tool lets you change menu costs immediately, making sure your online listing always matches your current financial model.

Real-World Applications

Handling a sudden rush of orders

A manager notices 15 new pending orders. Instead of logging into the tablet and hitting 'Accept' fifteen times, they prompt their agent: 'Review all pending orders and accept them for preparation.' The MCP handles the `accept_order` sequence instantly.

Coordinating a multi-site inventory check

A dark kitchen operator needs to know which brand's menu items are sold out across three different locations. They ask for a full list of menu sections, and the agent runs `list_menu_sections` on each venue to confirm stock levels before going live.

Emergency courier pickup

A logistics manager needs parts picked up from a client's site across town. They tell their agent: 'Request a courier for a package pick-up at X address and drop off at Y address.' The MCP handles the `request_courier` call, giving them a tracking ID.

Addressing pricing discrepancies

The finance team realizes the price of their flagship dish is wrong online. They tell the agent to 'Update the price for the main course' using `update_product_price`, ensuring the marketplace listing reflects the correct cost immediately.

Patterns to Avoid

Only checking order status

✗ AVOID

A user only asks, 'What orders do I have?' and then gets a list. They are left to manually check if items need price updates or if they need to request couriers.

✓ INSTEAD

To get the full picture, you must combine calls. Ask your agent: 'List all pending orders, check their status, and let me know if any courier pickups are needed.' This uses `list_orders` combined with `request_courier`.

Manually updating menus

✗ AVOID

A staff member has to log in, navigate through multiple menu categories, and manually toggle the status of sold-out items one by one.

✓ INSTEAD

Use `update_product_status` or `list_menu_sections`. Tell your agent: 'Mark all products under the Appetizer section as unavailable.' This saves hours of clicking.

Ignoring delivery logistics

✗ AVOID

The kitchen finishes an order and waits for a courier to show up, manually calling the dispatch line when they are ready.

✓ INSTEAD

Instead, tell your agent: 'Mark Order #1234 as ready.' This triggers `mark_order_ready`, initiating the automated courier assignment process.

The Right Fit

Use this MCP if your business needs programmatic control over the entire order and delivery lifecycle on PedidosYa. You need to handle real-time state changes, like accepting a sudden influx of orders or instantly marking products sold out based on inventory counts. However, don't use this if your only goal is simple reporting; for historical data retrieval across all venues, you might just need `list_restaurants` and `get_restaurant`. Also, if you only manage basic payments and don't care about courier tracking or menu status, a simpler payment gateway MCP will suffice. But when order flow, inventory control (`update_product_status`), and physical logistics are intertwined, this is your tool.

Managing restaurant operations used to be a mess of clicking through tabs.

Today, accepting orders requires opening the tablet app, finding the pending queue, tapping each order one by one, and confirming acceptance. If you run out of a popular side item, you have to navigate deep into the menu structure, find that specific product listing, and toggle it off so customers don't try ordering it—all while juggling other tasks.

With this MCP, the same process happens in plain English. You just tell your agent: 'Accept all pending orders,' or 'Mark out-of-stock items.' The platform does the mechanical work, giving you immediate control without touching a single screen.

PedidosYa MCP gives you instant menu and order management.

The manual steps that vanish are logging into multiple platforms to check inventory, manually updating prices after cost changes, or waiting for a dispatcher call when an order is ready. You lose the physical dependency on the Point of Sale terminal.

Now, your entire operation runs through natural language prompts. It’s immediate, comprehensive control over every stage, from menu listing via `list_menu_sections` to final delivery tracking using `track_shipment`. Period.

PedidosYa with 14 Tools

These tools give you direct access to key functions like order status checks, menu adjustments, and courier dispatching for full control over your delivery business.

#	TOOL	DESCRIPTION
01	accept_order	Accepts a pending customer order, signaling to the system that your kitchen is preparing the items for pickup.
02	update_product_price	Changes the listed price of any menu item, with the update taking effect immediately on the public listing.
03	update_product_status	Toggles a product's availability online, marking it sold out or bringing it back to the active menu without changing its price or description.
04	create_webhook	Sets up event-driven webhooks so your systems automatically react when things like new orders or cancellations occur.
05	get_order	Retrieves the complete details for a specific customer order using an identifier.
06	get_restaurant	Fetches operational data and status information for a particular registered restaurant location.
07	list_menu_sections	Gets a full list of menu categories and all products within them, including current pricing and availability.
08	list_orders	Lists incoming orders, allowing you to filter by status—pending, being prepared, or ready for pickup.
09	list_restaurants	Shows a list of all your partner restaurant venues, including their operational hours and current open/closed state.
10	list_webhooks	Displays all the webhook event subscriptions you have currently configured.
11	mark_order_ready	Triggers the courier dispatch process for an order, assuming a delivery courier hasn't arrived yet.
12	reject_order	Declines an incoming customer order and provides a specific reason like 'out of stock' or 'too busy'.
13	request_courier	Schedules an on-demand courier pickup for external deliveries, specifying both the starting and ending addresses.

#	TOOL	DESCRIPTION
14	track_shipment	Provides real-time GPS coordinates and status updates for a currently active courier shipment.

See It in Action

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

U Show me all pending orders for my restaurant.

You have 3 pending orders: 1. Order #PY-88412 — 2x Burger Clásica + 1x Papas Fritas ($345 UYU, 5 min ago). 2. Order #PY-88415 — 1x Pizza Margherita + 2x Coca-Cola ($520 UYU, 2 min ago). 3. Order #PY-88418 — 3x Empanadas + 1x Flan ($280 UYU, just now). Would you like me to accept all three?

U Mark the Chocotorta as sold out on restaurant ID R-4421.

Done! Product 'Chocotorta' (ID: P-9922) has been marked as unavailable on restaurant R-4421's PedidosYa menu. Customers will see it greyed out until you re-enable it. Would you like to update any other products?

U Request a courier to pick up a package from Av. 18 de Julio 1234, Montevideo and deliver to Rambla Wilson 500.

Courier requested! Shipment #SH-77201 has been created. Pickup: Av. 18 de Julio 1234, Montevideo. Dropoff: Rambla Wilson 500, Montevideo. Estimated pickup in 12 minutes. A courier will be assigned shortly. Use `track_shipment` with ID SH-77201 to monitor real-time GPS location.

Frequently Asked Questions

01 **How do I accept an order using the PedidosYa MCP?**

You use the `accept_order` tool. Simply tell your agent to 'Accept all pending orders' and it executes the necessary steps, committing you to preparing the items.

02 **Can I update prices with the PedidosYa MCP?**

Yes, use the `update_product_price` tool. You provide the menu product ID and the new local currency price, and it updates instantly on the live listing.

03 **What is the best way to track a delivery with PedidosYa MCP?**

Use the `track_shipment` tool. You input the shipment ID provided when you run `request_courier`, and the agent gives you real-time GPS updates on the courier's location.

04 **How do I handle sold out items with PedidosYa MCP?**

Use `update_product_status`. You specify the product ID, and it toggles the item off the menu instantly. This is much faster than manually editing the menu listing.

05 **Can I set up automatic alerts for new orders with PedidosYa MCP?**

Yes, you use `create_webhook`. You specify events like 'new_order' or 'order_cancelled,' and your systems will automatically get notified when those things happen.

06 **Can my AI agent accept or reject incoming orders automatically?**

Yes! Use `accept_order` to confirm an incoming order or `reject_order` with a reason (out_of_stock, closing_soon, too_busy, item_unavailable). Once accepted, the PedidosYa system begins assigning a delivery courier. When ready, use `mark_order_ready` to signal the courier for pickup.

07 **How do I mark items as sold out on my live menu?**

Use the `update_product_status` tool with the restaurant ID, product ID, and set `available` to false. The product will immediately show as unavailable on PedidosYa. To bring it back, call the same tool with `available` set to true.

08 **Can I request a courier for deliveries outside of food orders?**

Yes! Use the `request_courier` tool to create a shipment request with pickup and dropoff addresses. This leverages PedidosYa's logistics network for B2B courier services, independent of restaurant orders. Track the courier in real-time with `track_shipment`.

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 `"pedidosya": { "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

PedidosYa 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 PedidosYa. 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	PedidosYa MCP
Server ID	`019d846a-acca-707e-ba12-3b0468d4dd1a`
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/pedidosya.