

MCP SERVER

NO CODE

CLOUD HOSTED

# Odoo Purchase MCP

Manage your entire buying lifecycle with natural language.

Odoo Purchase handles your entire purchasing workflow directly through your AI agent. Use this MCP to search for vendors, draft new purchase orders (RFQs), check item costs on specific lines, and confirm those quotes into official, processed purchase orders. It lets you manage everything from initial supplier contact to final receipt creation without ever leaving your chat window.

**A+** Quality Score 100/100

purchase-orders

vendor-management

procurement

rfq-management

supplier-relations

inventory-replenishment



# 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](https://cloud.vinkius.com) — connect your AI agent in under 60 seconds.

# Odoo Purchase MCP

7 tools available

Cloud-hosted on Vinkius

Managing purchases in a large company usually means jumping between vendor portals, inventory systems, and the main ERP dashboard. This MCP fixes that. You tell your agent what you need—say, 'I need three different types of widgets from Acme Corp.' Your AI client connects to Odoo and finds the necessary supplier data, drafts the full quote request, and handles the initial paperwork. Once approved, it confirms the order, notifying the vendor and even setting up the expected warehouse receipt in one go. This centralized control means you spend time making decisions, not clicking buttons or copying IDs. Connecting this MCP via Vinkius gives your agent access to Odoo's entire procurement lifecycle, letting you manage supplier relationships and stock replenishment purely through conversation.

---

## Core Capabilities

### 01 — Search for Suppliers

Find vendor contact details like phone numbers and emails by searching the database.

### 03 — List Pending Quotes

Retrieve a list of all requests for quotation that are waiting for pricing or internal approval.

### 05 — Inspect Line Items

Check the detailed breakdown of products within a single purchase order, including unit cost and planned delivery dates per item.

### 02 — Initiate Purchase Requests

Create a new, preliminary purchase order draft (RFQ) specifying products, quantities, and costs for a vendor.

### 04 — Get Order Overview

View an overall status, total amount, and expected delivery date for any specific purchase order number.

### 06 — Finalize Order Confirmation

Approve a draft RFQ, converting it into an official Purchase Order that triggers vendor notifications and warehouse receiving records.

# One Click on Vinkius — From Prompt to Execution

Available at [vinkius.com/mcp/odoo-purchase](https://vinkius.com/mcp/odoo-purchase) — connect your AI agent in three steps.

- 01** First, tell your agent what you need. You might ask to find a supplier's contact info or list all pending quotes.
- 02** Next, the MCP uses Odoo's tools to pull data—maybe it creates an initial purchase order draft with product lines and costs for review.
- 03** Finally, when everything is correct, your agent confirms the transaction. The system locks the order, sends vendor notifications, and sets up the incoming receipt in the warehouse.

The bottom line is you use natural language to execute complex, multi-step ERP transactions that usually require several screens and clicks.

---

## Built For

This MCP is for anyone whose job involves acquiring physical goods or managing vendor relationships. You're the purchasing manager who gets frustrated logging into three different systems just to verify one order status, or the operations analyst who needs quick access to specific line item pricing details.

### **Purchasing Manager**

They use this MCP constantly to list all open purchase orders and confirm drafts into final, tracked records.

### **Operations Analyst**

They check detailed line items on existing POs to verify if the unit cost or expected delivery date matches current inventory projections.

### **Account Manager**

They quickly search for new vendor contact information and draft initial purchase requests without needing a dedicated database lookup tool.

## What Changes When You Connect

- 
- 01 Confirming orders is instant. Instead of logging into Odoo to manually change a status from 'draft' to 'purchase', you simply ask your agent, and the `odoo_confirm_purchase_order` tool handles the approval, vendor notification, and receipt creation automatically.

---

  - 02 Never start from scratch. If you need to buy something new, use `odoo_search_vendors` first to verify contact details, then use `odoo_create_purchase_order` to draft a request instantly with all product lines included.

---

  - 03 Track everything in one go. To see the status of multiple pending items, run `odoo_list_purchase_orders`. You get the PO number, vendor name, current state, and total amounts without changing tabs.

---

  - 04 Deep dives are easy. If you only need to check the costs or expected delivery dates for a few specific products on an order, use `odoo_get_po_lines` instead of pulling up the entire record.

---

  - 05 Manage pending work. If your team is waiting on pricing, run `odoo_list_rfqs`. This tool shows only the requests that are stuck in draft status and need someone's attention or negotiation.
- 

---

## Real-World Applications

### Approving a vendor quote

A purchasing manager gets an email with an approved RFQ. Instead of logging into Odoo, navigating to the correct draft record, and manually clicking 'Confirm,' they ask their agent to confirm it. The tool runs `'odoo_confirm_purchase_order'`, which locks in the order, alerts the vendor, and creates the incoming receipt for inventory.

### Verifying a large shipment

An operations analyst needs to know if the latest expected delivery date for three specific product lines—say, motors, cables, and valves—matches the quote. They use `'odoo_get_po_lines'` on the PO number; it returns all three details in one structured response.

### Checking outstanding requests

The team lead asks for a list of all purchase requests that are waiting on pricing from external vendors. The agent uses ``odoo_list_rfqs``, giving the lead an immediate, filtered view of everything stalled in 'draft' status.

### Finding new suppliers

A manager needs to buy something but doesn't have a vendor ID. They ask their agent to search for local electrical equipment providers. The tool uses ``odoo_search_vendors`` and returns the name, email, and phone number so they can start drafting the request.

---

## Patterns to Avoid

---

### Checking PO status manually

#### ✗ AVOID

The user has to remember the exact PO ID, navigate to the list of all orders, filter by date range, and then click on each result one by one just to see if it's 'sent' or 'draft'.

#### ✓ INSTEAD

Just ask your agent to run ``odoo_list_purchase_orders``. It gives you a summary table with the PO number, vendor name, status, and total amount all at once.

### Creating an order from scratch

#### ✗ AVOID

The user drafts the purchase request in one system but has to copy the vendor's email and product IDs into a completely different ERP interface.

#### ✓ INSTEAD

First, run ``odoo_search_vendors`` to get verified contact data. Then, use ``odoo_create_purchase_order``, feeding it the structured JSON array of products and costs directly.

### Misinterpreting order status

#### ✗ AVOID

A user sees an order is 'draft' but doesn't know if that means no one has seen it yet or if they just haven't approved it. They waste time chasing the wrong person.

#### ✓ INSTEAD

If you only want to see things needing internal approval, use ``odoo_list_rfqs``. This specifically filters for Requests for Quotation that are stalled and need action.

---

## The Right Fit

Use this MCP if your core job involves the physical process of buying goods—from initial quote gathering to final inventory receipt. If you spend time tracking vendor IDs, confirming quotes, or checking line item costs before an order goes through, this is for you. Don't use it if your main task is handling sales orders (that

requires a separate Sales MCP) or managing internal employee records (you need HR tools). For instance, if you only need to check general company contact data without linking it to a purchase process, the `odoo_search_vendors` tool works well. But if you are building an agent that needs to execute multi-step actions—like searching for vendors AND then confirming an order—you'll need this entire MCP connected via Vinkius.

---

## The Procurement Funnel is a Cluster of Tabs and Manual Updates

Today, buying things means juggling at least three different screens. You start by looking up vendor IDs in one system, then you build the quote draft on another platform, and finally, when it's time to approve, you have to manually switch over to the main ERP dashboard to change the status from 'draft' to 'purchase.' It's constant copy-pasting of numbers and chasing down which specific record needs attention.

With this MCP, that entire flow collapses into a single conversation. You describe the goal—'I need to approve and process the widget order for Acme Corp.' Your agent handles the lookup, drafts the necessary records via `odoo_create_purchase_order`, confirms it using `odoo_confirm_purchase_order`, and gives you a final summary of exactly what happened.

---

## Odoo Purchase: Tracking Orders Through Natural Language

The ability to pull the full details on any PO using `odoo_get_purchase_order` eliminates having to remember specific record IDs or navigate complex drop-down menus. You just ask, and you get the complete picture: who, how much, when.

What changes is that your agent acts as a single pane of glass for your entire supply chain process. It stops being about knowing where the data lives; it's only about getting the answer.

---

# Odoo Purchase: 7 Tools for ERP Operations

These tools give you granular control over every stage of the purchasing process in Odoo. You can list, create, search, and confirm orders using your agent.

#	TOOL	DESCRIPTION
01	<code>odoo_confirm_purchase_order</code>	Approves a preliminary quote, turning it into a confirmed order that notifies the vendor and sets up warehouse receiving records.
02	<code>odoo_create_purchase_order</code>	Drafts a brand new purchase request for a vendor using specified products, quantities, and unit costs.
03	<code>odoo_get_po_lines</code>	Retrieves the individual product line details of an order, showing cost, quantity, and estimated delivery dates.
04	<code>odoo_get_purchase_order</code>	Pulls all core data about one specific purchase order by its number for a complete status check.
05	<code>odoo_list_purchase_orders</code>	Lists multiple current and completed orders, giving you the vendor name, amounts, total status, and expected receipt date.
06	<code>odoo_list_rfqs</code>	Retrieves only draft requests for quotation that are waiting for pricing or internal approval from a vendor.
07	<code>odoo_search_vendors</code>	Finds and lists contact information, including email and phone number, for existing suppliers in the system.

---

## See It in Action

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

### **U** Search for leads from the website



#### **CRM Leads — Website**

NAME	EMAIL	STAGE	REVENUE
Acme Corp	info@acme.com	Qualification	\$15,000
Beta Inc	hello@beta.io	Proposition	\$8,500

### **U** Show recent sales orders



#### **Sales Orders**

SO#	CUSTOMER	AMOUNT	STATUS
S00042	Acme Corp	\$12,500	Confirmed
S00041	Beta Inc	\$3,200	Draft

---

## Frequently Asked Questions

### **01** How do I find vendor contact info using `odoo_search_vendors`?

You tell your agent to search for a supplier name. It runs ``odoo_search_vendors`` and returns their verified email, phone number, city, and country directly in the chat response.

### **02** Can I list all my pending RFQs with `odoo_list_rfqs`?

Yes. Running ``odoo_list_rfqs`` filters out every completed or confirmed order and shows only those Requests for Quotation that are currently waiting on pricing, negotiation, or final approval.

**03 What is the difference between `odoo_create_purchase_order` and `odoo_get_purchase_order`?**

Creating a purchase order uses `odoo_create_purchase_order` to build a new draft. Getting details uses `odoo_get_purchase_order` to retrieve all existing information about a single, specific PO number.

---

**04 Does `odoo_confirm_purchase_order` automatically notify the vendor?**

Yes, confirming an order with `odoo_confirm_purchase_order` is designed to do more than just change the status. It locks the order, notifies the external vendor, and creates a corresponding incoming receipt record for inventory.

---

**05 I need to check item costs on an existing PO; which tool should I use?**

Use `odoo_get_po_lines`. This function specializes in returning the individual line items, letting you inspect product names, ordered quantities, unit cost, and planned delivery dates per item.







---

# Go Live in 60 Seconds

Get your connection token from [cloud.vinkius.com](https://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
 <b>Claude AI</b>	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
 <b>Cursor</b>	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
 <b>VS Code</b>	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add <code>"odoo-purchase": { "url": "..." }</code>
 <b>Windsurf</b>	MCP Settings → <code>mcp_settings.json</code> → Add endpoint URL
 <b>ChatGPT</b>	Settings → Tools & plugins → Add MCP server → Paste endpoint
 <b>Gemini</b>	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

# Odoo Purchase 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](https://vinkius.com) · [support@vinkius.com](mailto: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 Odoo Purchase. 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	Odoo Purchase MCP
Server ID	019d75e3-037d-70ea-96aa-1dd49f33f043
Platform	Vinkius Cloud for AI Agents
Endpoint	<a href="https://edge.vinkius.com/{token}/mcp">https://edge.vinkius.com/{token}/mcp</a>

### 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/odoo-purchase](https://vinkius.com/mcp/odoo-purchase).