MCP SERVER | NO CODE | CLOUD HOSTED

Itemized Shared Grocery Splitter MCP for AI Agents

Accurate Expense Splitting and Household Bill Management

Use Itemized Shared Grocery Splitter MCP when you need to accurately divide a grocery bill among multiple people based on what each person actually took home. It calculates individual item costs before tax and then adds an even share of the total sales tax, giving you precise final balances for roommates or friends.

A+ Quality Score 100/100

grocery | splitting | receipt | expenses | shared-costs

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JULY 2026
vinkius.com/mcp/itemized-shared-grocery-splitter

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.

Itemized Shared Grocery Splitter MCP

3 tools available
Cloud-hosted on Vinkius

Splitting receipts used to be a nightmare of manual math—you'd track who bought which carton of eggs, calculate subtotals, and then figure out how to divide the sales tax fairly. This MCP solves that mess completely. It lets your AI client process a shared receipt and instantly determine exactly what everyone owes. You can first identify every person involved in the shopping trip. Then, it calculates individual sub-totals for specific items consumed by one or more people. Finally, it computes the total final balance owed by each participant, ensuring that tax is split evenly across the group. It's perfect for anyone managing shared household costs; just connect this MCP through Vinkius and let your agent handle the math.

Core Capabilities

01 — Identify trip participants

The MCP pulls a list of every unique person involved in the shopping trip from the receipt.

02 — Calculate item-specific costs

It figures out how much each individual owes for specific food items before any taxes are applied.

03 — Determine final shared balances

The system computes the total amount each person must pay, adding an equal share of the total tax to their item subtotal.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/itemized-shared-grocery-splitter** — connect your AI agent in three steps.

01 Provide the AI client with the shared grocery receipt data.

02 The MCP first identifies every unique shopper on the trip. Next, it calculates the specific cost for each person based only on items they consumed. Finally, it computes the final amount due by adding an equal share of the total tax to everyone's calculated subtotal.

The bottom line is you get a clear breakdown of who owes what, including accurate itemization and shared tax costs.

Built For

This MCP is built for people managing joint living expenses or group purchases. If you're tired of arguing over whose turn it was to track the receipts or doing math on a napkin, this tool saves your sanity. It's essential for roommates and friends who share household costs.

Roommates

Uses the MCP to settle bills after joint shopping trips, ensuring every item is properly assigned before splitting tax.

Group Trip Organizer

Processes complex group receipts from events or outings, accurately assigning costs and calculating final payments for all attendees.

Friends/Flatmates

Settles shared grocery bills without the hassle of spreadsheets. It provides immediate clarity on who owes what to whom.

What Changes When You Connect

01 Stop guessing who owes what. By using calculate_individual_subtotals, you get a precise breakdown of item costs before tax, eliminating disputes.

02 Tax division is simple with compute_final_balances. It automatically adds an equal share of the total sales tax to everyone's bill, making settlement quick and fair.

03 You save time on manual calculations. The MCP handles all the complex math—identifying participants, itemizing costs, and calculating final balances—in minutes.

04 Clarity for shared living. Instead of confusing spreadsheets, you get a simple, clear report showing exactly what each person needs to pay back after every shopping trip.

05 Works with your favorite AI client. You connect this MCP via Vinkius and let your agent manage the entire financial workflow without needing specialized accounting software.

Real-World Applications

Settling bills after a weekend grocery run

A group of four roommates finished shopping. Instead of dividing everything by four, the user asks their agent to calculate final balances using the MCP. The agent identifies all participants and returns a list showing each person's exact share, including tax.

Accounting for shared household supplies

The user needs to settle costs among family members for a joint supply run. The agent uses the MCP to identify every unique participant and then computes the final balance, ensuring tax is correctly distributed across all parties.

Figuring out who bought which snacks on vacation

A friend brings home receipts from an outing with multiple purchases. The user inputs the data and asks the agent to calculate individual subtotals for specific items consumed by certain people, making sure no one pays for another's chips.

Patterns to Avoid

Using simple percentage splits

✗ AVOID

Simply dividing the total bill by the number of people ignores the fact that some items were only for one person. This leads to inaccurate over- or underpayment.

✓ INSTEAD

Instead, use Itemized Shared Grocery Splitter MCP. First, let your agent get_unique_participants to list everyone. Then, use calculate_individual_subtotals before running compute_final_balances.

Manual tracking in spreadsheets

✗ AVOID

Creating complex formulas to track subtotals and then manually distributing tax is time-consuming and prone to human error.

✓ INSTEAD

Let your agent run the MCP. It handles the full sequence: get_unique_participants, calculate_individual_subtotals, and compute_final_balances—all in one workflow.

Ignoring tax distribution

✗ AVOID

Calculating individual item shares but forgetting to account for sales tax leads to an incomplete settlement that leaves someone paying too little or too much.

✓ INSTEAD

Always finish the process by calling compute_final_balances. This tool guarantees the final total includes an equal share of the overall tax for everyone.

The Right Fit

Use this MCP if your expense splitting needs to be granular: you must split costs item-by-item and distribute sales tax equally among participants. It's perfect when household goods or groceries are involved, and tracking usage is key. Don't use it if you just need to split a single, lump-sum bill (like rent) among people; then a simple percentage calculation works fine. If your expenses involve complex services outside of groceries (like travel tickets or utilities), look for a different financial MCP that handles those specific categories.

Itemized Shared Grocery Splitter: Solving Expense Splitting with Itemization

Right now, settling bills feels like an archaeological dig. You're staring at a long receipt, trying to mentally cross-reference who picked out the cereal and who grabbed the fancy cheese. You end up opening multiple tabs—a calculator here, a notes app there—and spending way too much time arguing over whether that bulk item should be split evenly or by weight.

With this MCP, your agent handles it all. It processes the receipt to identify every person involved and calculates individual subtotals for specific items instantly. You don't spend the weekend reconciling costs; you just get a clear, accurate list of what everyone actually bought.

Itemized Shared Grocery Splitter: Managing Tax Allocation in Household Bills

The trickiest part is often the tax. You calculate who owes for milk and bread, but then you have to figure out how to split that general sales tax money fairly. Most people just approximate it or skip it entirely, leading to a few dollars of dispute every time.

This MCP fixes the final step with compute_final_balances. It takes everyone's item share and adds an equal portion of the total tax back in. The result is always mathematically sound, keeping your shared finances clean.

Itemized Shared Grocery Splitter: 3 Tools for Expense Splitting

Use these tools to identify participants, separate item costs from tax, and compute the final payment amount for shared receipts.

#	TOOL	DESCRIPTION
01	`calculate_individual_subtotals`	Determines how much each person is responsible for based solely on the food items they ate, excluding tax.
02	`compute_final_balances`	Calculates the total final amount required from each participant by adding their share of the sales tax to their item subtotal.
03	`get_unique_participants`	Identifies and lists every distinct person who was involved in the shopping trip based on the provided receipt data.

See It in Action

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

U Hey, can you figure out who shopped today? Based on this receipt: [{'name': 'Eggs', 'price': 5.00, 'consumedBy': ['Me', 'Mom']}, {'name': 'Wine', 'price': 25.00, 'consumedBy': ['Dad']}]?

The unique participants involved in the shopping trip are:

- **Me**
- **Mom**
- **Dad**

This list is based on all items found on the receipt.

U I need to know how much Mom owes for this grocery haul, including tax. Items: [{'name': 'Chicken Breast', 'price': 10.00, 'consumedBy': ['Mom']}, {'name': 'Rice', 'price': 5.00, 'consumedBy': ['Me']}] and total tax was $2.00.

Here's the breakdown for Mom:

Subtotal (Items): $10.00 (for Chicken Breast)
Tax Share: $1.00 (half of $2.00)

Total Due from Mom: $11.00

U What's the final balance for Dad? Items: [{'name': 'Bread', 'price': 3.50, 'consumedBy': ['Dad']}] with a total tax of $1.50.

The participants are just Dad in this case.

Calculation:

- Item Subtotal (Bread): $3.50
- Tax Share: $0.75 (half of $1.50)

Dad owes a final balance of $4.25.

Frequently Asked Questions

01 **How does Itemized Shared Grocery Splitter help me calculate split costs?**

It solves the math problem for you by calculating individual subtotals based only on what each person consumed. It then adds an equal share of sales tax to give a final, accurate payment amount.

02 **Can I use Itemized Shared Grocery Splitter if we have more than two people?**

Yes, it handles any number of participants. The MCP first identifies every unique person on the receipt so that everyone is correctly included in the final balance calculation.

03 **What about splitting sales tax with Itemized Shared Grocery Splitter?**

It ensures fairness by taking the total tax amount and dividing it equally among all participants. This prevents one person from accidentally paying or receiving too much tax.

04 **Is this better than just dividing the total bill evenly?**

Absolutely. Dividing the total is inaccurate because it doesn't account for itemization. This MCP calculates subtotals by specific items, which gives you a precise record of who bought what.

05 **Does Itemized Shared Grocery Splitter work for non-grocery expenses?**

It is primarily designed for shared goods receipts (groceries). For other types of bills, like rent or utilities, you'd need a different type of expense splitting MCP.

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 `"itemized-shared-grocery-splitter": { "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

Itemized Shared Grocery Splitter 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 Itemized Shared Grocery Splitter. 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	July 2026
MCP Server	Itemized Shared Grocery Splitter MCP
Server ID	`019f1fe1-5786-7246-a836-e8e1652eaf06`
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/itemized-shared-grocery-splitter.