MCP SERVER | NO CODE | CLOUD HOSTED

Uneven Income Splitter MCP for AI Agents

Calculating Fair Proportional Expense Sharing for Joint Bills

The Uneven Income Splitter MCP calculates fair expense sharing by distributing bills proportionally based on individual income levels. Stop dividing shared costs equally when some people earn significantly more than others. This tool takes the total bill amount and a roster of participants, factoring in their incomes to give everyone an exact monetary share. It handles complex math—including percentage breakdowns and payment verification—ensuring financial fairness whether you're splitting rent or organizing a group trip.

A+ Quality Score 100/100

finance | split-bill | proportional | income-based | expense-sharing

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JULY 2026
vinkius.com/mcp/uneven-income-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.

Uneven Income Splitter MCP

3 tools available
Cloud-hosted on Vinkius

Splitting bills when your friends have wildly different incomes is always messy. You end up arguing over what 'fair' even means, often resorting to rough estimates that feel wrong. The Uneven Income Splitter changes that. Instead of dividing the cost by four (the simple but inaccurate method), this MCP uses everyone’s income as a weight to calculate individual responsibility. For example, if one person earns five times more than another, they should pay five times more toward shared expenses.

Using your AI client connected through Vinkius, you feed it the total bill and the list of incomes. The system instantly breaks down who owes what. It doesn't just give a guess; it provides an exact financial breakdown for every single person. You can even audit payments later on to make sure everyone paid exactly their calculated share, eliminating those annoying rounding discrepancies that always pop up at the end of the month.

This tool takes the guesswork out of group finance, letting you move past arguing and straight to enjoying the shared expense it was for.

Core Capabilities

01 — Calculate precise expense splits

Determines the exact dollar amount each person must pay toward a total bill based on their income relative to the group.

02 — Determine individual contribution percentage

Calculates what exact weight or percentage of the collective responsibility belongs to a single participant.

03 — Audit payments for financial accuracy

Compares submitted payments against expected proportional shares, flagging any discrepancies in the group's finances.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/uneven-income-splitter** — connect your AI agent in three steps.

01 Provide your AI client with three pieces of data: the total bill amount, and a list of all participants along with their individual incomes.

02 The MCP calculates each person's precise percentage contribution based on their income compared to the group's combined income.

03 It delivers an exact monetary breakdown for every participant, telling you exactly what they owe toward the shared expenses.

The bottom line is that you tell your agent who paid what and who earned what; this MCP tells you if it all adds up fairly.

Built For

This tool is essential for anyone dealing with shared finances where income disparity makes equal splitting unfair. If you organize group trips, manage joint rent payments, or handle family expenses, this MCP stops the arguing and provides mathematical certainty.

Roommate/Tenancy Coordinator

Uses it monthly to split utilities, internet, and rent among housemates who have varying incomes. They get a precise breakdown instead of splitting everything 50/50.

Travel Group Organizer

Before or after a trip, they feed the MCP the total cost (flights, hotels) and everyone's income to ensure costs are split fairly based on who is better positioned to pay.

Financial Consultant/Bookkeeper

Uses it to audit shared financial agreements or partnership expenses, ensuring that the proposed payments match established proportional rules.

What Changes When You Connect

01 You stop arguing over rounding errors. The `verify_payment_accuracy` tool automatically checks payments against expected proportional shares, guaranteeing financial integrity.

02 The system moves beyond simple equal division. It uses income levels to calculate the true weight of each person's contribution, giving you a mathematically sound split.

03 You get instant clarity on who pays what. Using `get_propose_split` provides an exact monetary breakdown for every participant with minimal input from your agent.

04 You quantify fairness instantly. The ability to calculate the individual share percentage tells you exactly how much influence each person's income has on the total cost.

05 The process is streamlined. Instead of manually calculating ratios and percentages, your AI client handles the complex math automatically.

Real-World Applications

Roommates splitting utility bills

A group needs to split a $400 utility bill. They feed their agent the total cost and everyone's incomes. The MCP returns specific dollar amounts, showing that Person A (higher income) owes significantly more than Person B (lower income), eliminating disputes over who pays what.

Organizing a joint trip to Europe

The total cost of the trip is $8,000. The organizer uses the MCP with everyone's incomes. It generates an exact split showing that each person contributes according to their financial capacity, making the shared finances clear upfront.

Auditing partnership expenses

A small business partner submits receipts and payments for a $12,000 joint marketing expense. The user runs `verify_payment_accuracy` to confirm that every payment aligns perfectly with the initial income-based split agreement.

Calculating family contribution after an emergency

A large unexpected bill hits. The family uses the MCP, inputting the total cost and each member's current financial status to determine a fair, proportional payment plan that everyone can actually afford.

Patterns to Avoid

Splitting bills equally

✗ AVOID

The user assumes all shared costs should be split 4 ways ($100 each), even though one person earns ten times more than the others. This causes financial resentment and disputes.

✓ INSTEAD

Instead, use `get_propose_split`. Input the total bill and everyone's actual incomes. The MCP calculates a weighted share, ensuring the division reflects real-world income disparities.

Ignoring payment discrepancies

✗ AVOID

The group tracks payments manually in a spreadsheet but fails to account for small rounding errors or missed contributions, leading to an inaccurate final balance.

✓ INSTEAD

Run `verify_payment_accuracy`. This tool audits the submitted payments against the expected proportional shares, catching any mathematical inconsistencies and keeping your finances clean.

Using general expense trackers

✗ AVOID

A standard budgeting app tracks who paid what but doesn't calculate *if* that payment was fair based on income. It only records the transaction.

✓ INSTEAD

You must use this MCP. It calculates fairness first, then figures out the specific dollar amount using `get_propose_split`. It's a calculation engine, not just a ledger.

The Right Fit

Use Uneven Income Splitter if your group expenses are based on differing levels of financial capacity. This MCP is perfect for roommates splitting utilities or organizing large joint trips where 'equal' division isn't fair. Don't use it if you simply need to track who paid which specific receipt; a basic ledger tool will work better. Also, don't rely on it if the underlying income figures are wrong—the

output is only as accurate as the inputs. If your primary goal is just tracking payments without needing proportional checks, using a standard accounting spreadsheet is simpler. However, if you need to confirm that actual payments match a complex, income-based split, this MCP's `verify_payment_accuracy` tool is essential.

Uneven Income Splitter: Ending Financial Arguments with Proportional Billing

Right now, splitting joint expenses—whether it's rent or a trip to the coast—is manual and stressful. You open a spreadsheet, throw in everyone's income side-by-side with the total bill, and then you start arguing about what 'fair' means. You end up doing back-and-forth calculations that are prone to rounding errors and always feel like an estimate.

With this MCP, your agent handles it all. You provide the total cost and the list of incomes. The system instantly calculates a precise breakdown, telling you exactly who owes what based on their actual financial weight in the group. It provides immediate, definitive answers, not just estimates.

Uneven Income Splitter: Ensuring Payment Accuracy for Shared Finances

The worst part isn't calculating the split; it's auditing it. You track payments over weeks, and when you finally tally up who paid what, you have to manually compare those receipts against the original proportional math. It's a tedious cross-check that takes forever.

This MCP solves that with `verify_payment_accuracy`. After everyone pays their shares, you run this tool. It audits all the provided payments instantly, confirming they match the required proportional logic and flagging any discrepancies before they become actual financial problems.

Uneven Income Splitter: 3 Tools for Accurate Financial Splitting

These tools allow your agent to calculate precise monetary shares, determine individual contribution percentages, and audit group payments against established proportional logic.

#	TOOL	DESCRIPTION
01	`calculate_individual_share_percentage`	Determines what percentage of the total group responsibility falls on a specific person based on their income.
02	`get_propose_split`	Calculates the precise dollar amount each individual must pay toward a specific shared bill using proportional logic.
03	`verify_payment_accuracy`	Audits multiple submitted payments against an expected total and confirms they match the group's proportional financial logic.

See It in Action

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

U Split a $600 utility bill for three roommates: Alex (income: 45k), Ben (income: 25k), and Chris (income: 80k).

💰 **Proportional Split Calculation**

Based on your combined incomes, the split is calculated as follows:

- **Chris:** $360.00 (Highest income)
- **Alex:** $180.00
- **Ben:** $60.00 (Lowest income)

Total: $600.00

This ensures the cost reflects individual earning power.

U Check if these payments for a $15,000 joint project are accurate: A paid 7k, B paid 3k, and C paid 5k. Their incomes are 40k, 20k, and 90k.

✅ **Payment Verification Status**

- Expected Total Share: $15,000.00
- Payment Check: **FAIL**
- Calculated Proportional Split:

* A owes: $7,000.00
* B owes: $3,500.00
* C owes: $4,500.00

- Discrepancy Found: B paid too little; they are short $500.

U What percentage of the group's total responsibility does my income represent if I earn 70k and the others earn 30k and 20k?

📊 Individual Contribution Analysis

PARTICIPANT	INCOME	PERCENTAGE SHARE
You (A)	$70,000	**58.3%**
B	$30,000	25.0%
C	$20,000	16.7%

Your income represents over half of the group's total weight.

Frequently Asked Questions

01 **How does Uneven Income Splitter handle complex expense sharing scenarios?**

It calculates expenses proportionally, meaning people pay according to their ability to contribute relative to others. You just need to provide the total bill and everyone's incomes.

02 **Can I use Uneven Income Splitter for splitting rent among roommates?**

Absolutely. If some housemates earn significantly more than others, you can feed those specific incomes into this MCP to get a fair, weighted division of the total monthly cost.

03 **What if I want to verify past payments using Uneven Income Splitter?**

You can audit previous payments. Just input the initial bill amount and all recorded payments; the MCP checks them against proportional logic to confirm financial accuracy.

04 **Is Uneven Income Splitter only for dollars, or does it work with other currencies?**

The tool is designed for monetary calculations. While you input amounts in a specific currency, the focus remains on maintaining accurate proportional ratios regardless of the unit.

05 **Does Uneven Income Splitter tell me who owes what dollar amount?**

Yes. It gives you an exact, definitive monetary breakdown for every single person involved in the shared expense, based on their income level.

06 **I don’t know how to use Uneven Income Splitter with my AI agent.**

It’s simple. Just ask your agent a natural language question: 'Split $2000 between me (50k) and John (10k).' The MCP handles the complex math for you.

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 `"uneven-income-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

Uneven Income 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 Uneven Income 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	Uneven Income Splitter MCP
Server ID	`019f1fe3-b540-7180-88a6-cf7a64262f61`
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/uneven-income-splitter.