MCP SERVER | NO CODE | CLOUD HOSTED

Cooking Session Timeline MCP

Coordinate complex multi-dish meals without timing errors.

Cooking Session Timeline MCP manages the math behind your kitchen timing. It sequences prep work, active cooking, and resting periods so every component hits the table at the right moment. It accounts for your specific kitchen constraints, like burner count and oven limits, to build a realistic master schedule for your meal.

A+ Quality Score 100/100

cooking | scheduling | culinary | optimization | planning

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · OCTOBER 2026
vinkius.com/en/ai-agent-connect/cooking-session-timeline

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.

Cooking Session Timeline MCP

4 tools available
Cloud-hosted on Vinkius

You shouldn't have to guess when to pull the roast out of the oven to ensure the sides are hot. This MCP handles the heavy lifting of culinary logistics. It looks at every dish you want to make, calculates the necessary prep time, and maps out the active cooking windows. It even factors in the mandatory resting periods for proteins so you aren't serving dry meat. If you're working with limited equipment, the MCP checks your plan against your actual hardware. It makes sure you don't accidentally schedule three dishes for the oven when you only have one. Instead of staring at a recipe and hoping for the best, you get a logical, step-by-step execution plan that respects your kitchen's limits.

Core Capabilities

01 — Master Scheduling

Your agent creates a single timeline that aligns all cooking and resting stages.

02 — Workload Balancing

The AI reorders prep tasks to prevent you from being overwhelmed or standing around idle.

03 — Hardware Validation

The MCP checks if your plan exceeds your available burners or oven capacity.

04 — Template Access

Your agent pulls from pre-set dish templates to speed up the planning process.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/en/ai-agent-connect/cooking-session-timeline** — connect your AI agent in three steps.

01 Connect the MCP to Claude, Cursor, or Windsurf via Vinkius.

02 Tell your agent what dishes you want to make and your serving time.

03 Provide your kitchen constraints like burner count and oven temperature.

04 The agent uses the tools to build, optimize, and validate your timeline.

Connect the MCP to your preferred client and start planning your menu.

Built For

This is for anyone managing complex meal service where timing is the difference between a great meal and a disaster.

Home Cooks

Users who want to host multi-course dinners without the stress of timing.

Culinary Students

Learners practicing how to manage multiple components in a professional kitchen setting.

Private Chefs

Professionals needing to quickly map out a logical workflow for specific client requests.

What Changes When You Connect

01 Prevents hardware conflicts by checking burner and oven availability.

02 Reduces idle time by reordering prep tasks for better flow.

03 Ensures all food is ready to serve at the same time.

04 Accounts for necessary resting periods for proteins.

Real-World Applications

Holiday Dinners

Coordinate a turkey, stuffing, and multiple vegetable sides so everything is hot at once.

Dinner Parties

Manage the timing of appetizers and main courses to keep guests fed on schedule.

Kitchen Efficiency

Reorder prep work to maximize your movement in a small kitchen space.

Resource Management

Verify that your planned menu fits within the limits of your single oven or limited burners.

Patterns to Avoid

Ignoring hardware limits

AVOID

You plan a massive feast with four roasted dishes but only have one oven. You assume you can just rotate them as you go.

✓ INSTEAD

Run your plan through `validate_resource_conflict` first. It catches these impossible schedules before you start preheating.

Manual prep sequencing

AVOID

You try to eyeball the prep work for five different sides, leading to long periods of standing around waiting for water to boil.

✓ INSTEAD

Use `optimize_prep_work` to reorder your tasks. It keeps you moving and eliminates those awkward gaps in your workflow.

Starting from scratch

AVOID

You spend an hour typing out every single step for a standard roast chicken dinner.

✓ INSTEAD

Pull existing structures using `get_dish_templates` to get a head start on your planning.

The Right Fit

Connect this MCP if you are managing multi-course meals or complex dinner parties where timing is critical. Do not use it if you are just making a single sandwich or a simple one-pot meal. The deciding factor is whether your menu requires synchronized timing and hardware management.

Cooking Session Timeline Tools

These tools handle the math and logistics of your kitchen, from building master schedules to checking your burner capacity.

#	TOOL	DESCRIPTION
01	`calculate_timeline`	This tool builds a complete, synchronized master schedule for all your planned dishes.
02	`get_dish_templates`	Use this to pull up pre-defined dish templates that serve as a starting point for your planning.
03	`optimize_prep_work`	This tool reorders your non-cooking tasks to cut down on idle time and keep you moving efficiently.
04	`validate_resource_conflict`	This tool checks your proposed sequence against your available burners and oven space to catch impossible plans.

See It in Action

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

U Generate a cooking timeline for a Roast Beef and Mashed Potatoes to be served at 7:00 PM. My oven goes up to 250°C and I have 4 burners.

6:00 PM: Start Prep for Roast Beef. 6:30 PM: Roast Beef enters oven at 190°C. 6:45 PM: Prep Mashed Potatoes. 7:00 PM: Roast Beef finishes cooking and begins resting. 7:05 PM: Serve Roast Beef and Mashed Potatoes.

U Check if this plan is feasible: Roast Beef on burner at 6:00 PM and Salmon on burner at 6:00 PM, but I only have 1 burner.

No, the plan is not feasible. There is a conflict: two dishes are assigned to a burner at 6:00 PM, but only 1 burner is available.

U What dish templates are available for proteins?

Available protein templates include Roast Beef and Salmon.

Frequently Asked Questions

01 **How does this MCP handle kitchen constraints?**

It uses the validate_resource_conflict tool to check your plan against the number of burners and oven capacity you provide.

02 **Can I use this with Claude or Cursor?**

Yes, you can connect this MCP to any MCP-compatible client like Claude, Cursor, or Windsurf.

03 **Does it help with prep work?**

Yes, the optimize_prep_work tool reorders your non-cooking tasks to minimize idle time.

04 **What if I have a limited number of burners?**

You can specify your burner count, and the MCP will ensure your schedule doesn't assign more dishes to a burner than you actually have.

05 **Does it include resting time for meat?**

Yes, the timeline includes the necessary resting periods to ensure food is served at its best.

06 **How does the tool handle oven constraints?**

The `calculate_timeline` tool checks your specified `ovenTemperatureLimit`. If a dish requires a temperature higher than your oven can reach, the calculation will fail to prevent invalid plans.

07 **Can I optimize my preparation time?**

Yes, you can use `optimize_prep_work` to reorder preparation tasks. You can choose between 'minimumDuration' for the fastest finish or 'balancedLoad' to spread out the work.

08 **How are stovetop burners managed?**

You specify your `burnerCount` during timeline generation. The engine ensures that the number of dishes using a burner at any single timestamp never exceeds your available capacity.

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 `"cooking-session-timeline": { "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

Cooking Session Timeline 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 Cooking Session Timeline. 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	October 2026
MCP Server	Cooking Session Timeline MCP
Server ID	`01a0f325-d039-71bc-a9a6-cbba33374d8e`
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/en/ai-agent-connect/cooking-session-timeline.