

M C P S E R V E R

N O C O D E

C L O U D H O S T E D

Editorial Deadline Calendar MCP

Build precise writing schedules that actually respect your workdays.

Editorial Deadline Calendar MCP handles the math behind your publishing timelines. It works backward from your fixed milestones to calculate exactly when tasks need to start, while automatically adjusting for holidays, weekends, and specific blackout dates. Your AI client uses these calculations to keep your editorial team on track without manual calendar math.

A+ Quality Score 100/100

scheduling

editorial

timeline

project-management

deadlines



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 — connect your AI agent in under 60 seconds.

Editorial Deadline Calendar MCP

4 tools available

Cloud-hosted on Vinkius

Managing a publishing schedule is a constant battle against the clock. You know the drill: a deadline is set, but you have to account for lead times, weekends, and those unexpected holidays that throw everything off. This MCP takes that math off your plate. Instead of manually counting days, you tell your AI client when a piece needs to go live, and it works backward to tell you when the research, drafting, and editing phases must begin.

It doesn't just count days; it understands your specific constraints. If you have a blackout period where no work happens, the MCP factors that in so your deadlines don't slip. It also spots when you've overbooked your team or proposed a timeline that is physically impossible to meet. You can use it to check if a new project fits into your current workflow or to see how a sudden holiday will push back your entire publishing queue. It's about moving from guesswork to a schedule that actually works.

Core Capabilities

01 — Backward Scheduling

Your agent calculates task start dates by working backward from a fixed deadline.

02 — Blackout Date Integration

The MCP adjusts timelines to account for holidays and non-working days.

03 — Conflict Detection

Your AI client identifies when multiple tasks overlap and create scheduling conflicts.

04 — Feasibility Checking

The agent verifies if a set of milestones can realistically be met given the required lead times.

05 — Workday Awareness

The tool calculates timelines based on specific workweeks and excludes non-working days.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/en/ai-agent-connect/editorial-deadline-calendar — connect your AI agent in three steps.

- 01 Connect your preferred MCP-compatible client to Vinkius.
- 02 Provide your deadline, lead times, and workweek constraints to your AI client.
- 03 The MCP calculates the schedule by working backward from your milestones.
- 04 Your agent delivers a chronological task list or feasibility report.

Connect your client to Vinkius and start scheduling immediately.

Built For

This MCP is built for editorial teams and content managers who need to turn fixed publishing dates into actionable task lists.

Editorial Managers

They hand off publishing milestones to the AI to generate team task lists.

Content Strategists

They use the tool to check if new content pushes are feasible within existing timelines.

Project Managers

They use the MCP to identify resource collisions and holiday impacts on production.

What Changes When You Connect

- 01 Calculates task start dates automatically based on lead times.
- 02 Accounts for holidays and blackout dates to prevent scheduling errors.

-
- 03** Identifies impossible timelines before they cause missed deadlines.
-
- 04** Spots resource collisions where tasks overlap too heavily.
-

Real-World Applications

New Project Planning

You provide a final launch date and lead times to get a full list of start and end dates for every phase.

Holiday Impact Analysis

You check how much a long holiday weekend will push back your upcoming publication dates.

Resource Management

You use the tool to see if your writers are being assigned too many tasks during the same window.

Timeline Validation

You verify if a tight deadline is actually achievable given the required workdays and existing blackout dates.

Patterns to Avoid

Ignoring holiday delays

❌ AVOID

You set a hard deadline for a Friday launch without accounting for the Monday holiday. Your team misses the deadline because they didn't have enough working days.

✓ INSTEAD

Use ``get_blackout_impact`` to see exactly how much a holiday pushes your schedule back before you commit to a date.

Overloading your writers

❌ AVOID

You assign three major articles to the same editor for the same week. You assume they can handle the volume without checking their current load.

✓ INSTEAD

Run ``find_task_collisions`` to identify when your team has too many tasks scheduled at once.

Setting impossible milestones

❌ AVOID

You promise a client a three-day turnaround for a project that requires five days of research and drafting. You ignore the required lead times.

✅ INSTEAD

Check your proposed dates with ``validate_timeline_feasibility`` to ensure the schedule is actually achievable.

The Right Fit

Editorial managers and content strategists should connect this MCP if they manage complex publishing queues with fixed dates. Teams with simple, single-task workflows won't find much value here. The deciding factor is whether your schedule depends on specific lead times and non-working days.

Editorial Deadline Calendar Tools

These tools handle the math of backward scheduling, blackout date impacts, and timeline validation.

#	TOOL	DESCRIPTION
01	find_task_collisions	This tool identifies periods where your team has too many tasks scheduled at the same time. It helps you spot resource over-allocation before it becomes a problem.
02	validate_timeline_feasibility	This tool checks if your proposed milestones and lead times are actually possible. It flags schedules that cannot be completed within the given timeframe.
03	get_blackout_impact	This tool measures how much specific blackout dates are delaying your overall editorial schedule. It quantifies the delay caused by holidays or non-working periods.
04	get_deadline_schedule	This tool generates a full, chronological list of tasks. It provides the specific start and end dates for every step in your workflow.

See It in Action

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

U Generate a schedule for a project with a final deadline on October 15th, requiring 5 workdays of lead time, assuming a Monday-Friday work week.



The task will start on October 8th and end on October 15th, accounting for the 5 required workdays.

U Is it possible to complete a 10-day task by Friday if today is Monday and there is a blackout date on Wednesday?



No, the timeline is infeasible because the 10-day lead time plus the Wednesday blackout date would require the task to start before the available workdays allow.

U How much did the Christmas holiday delay my publishing schedule?



The Christmas blackout period delayed the 'Publication Launch' milestone by 3 days.

Frequently Asked Questions

01 How does this MCP handle holidays?

The MCP uses blackout dates to adjust your timeline. It ensures that tasks are only scheduled on valid workdays, pushing start dates earlier if a holiday falls within a lead time.

02 Can I use this with Claude or Cursor?

Yes. You can connect this MCP to any MCP-compatible client, including Claude, Cursor, and Windsurf.

03 What happens if a deadline is impossible to meet?

The `validate_timeline_feasibility` tool will flag the schedule as infeasible. It tells your agent that the required lead times cannot be met within the available workdays.

04 Does it work with custom workweeks?

Yes. The scheduling logic accounts for specific workdays, allowing you to define when work actually happens.

05 How do I see if my team is overbooked?

You can use the `find_task_collisions` tool. It identifies specific periods where multiple tasks are scheduled to occur at the same time.

06 How does the tool handle holidays?

You can provide a list of blackout dates. The `'get_deadline_schedule'` tool will skip these dates when calculating start dates to ensure the timeline remains realistic.

07 Can I check if my editorial plan is actually possible?

Yes, use the `'validate_timeline_feasibility'` tool. It will analyze your milestones and lead times against your workdays and blackouts to identify logical impossibilities.

08 What happens if two tasks are scheduled at the same time?

You can use `'find_task_collisions'` to detect these overlaps. This helps you identify periods where your team might be over-allocated.

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 <code>"editorial-deadline-calendar": { "url": "..." }</code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → 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

Editorial Deadline Calendar 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 Editorial Deadline Calendar. 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	Editorial Deadline Calendar MCP
Server ID	01a0f60b-950d-7091-87ec-9dbbbaed7710
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/editorial-deadline-calendar.