MCP SERVER | NO CODE | CLOUD HOSTED

# Frame.io MCP

## Manage assets, comments, and project status conversationally.

Frame.io lets your AI agent manage video assets, track project progress, and handle team feedback without you leaving your chat window. Use this MCP to list projects, retrieve metadata, check collaborators, or add time-coded comments directly from Claude, Cursor, or any compatible client.

A+ Quality Score 100/100

video-review | creative-workflow | asset-management | media-collaboration | time-coded-comments | post-production

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/frameio

SCAN TO VIEW

# The infrastructure that powers AI agents in the real world.

Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

# Your AI Connections Run Through Vinkius Cloud

The world's largest managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect 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.

# Frame.io MCP

12 tools available
Cloud-hosted on Vinkius

Video review used to be a mess of email threads, shared spreadsheets, and scattered comment marks. Now, you can keep all that work centralized by connecting Frame.io via Vinkius. This MCP lets your AI agent read the project status, pull detailed asset metadata, or even add specific feedback directly into the video timeline—all through natural conversation.

Instead of clicking between tabs to find who owns an asset or checking a dozen email chains for approval status, you just ask. Your agent can list all projects in your team and instantly give you a summary of what's ready for review. It'll also pull up the full roster of collaborators involved. Whether you need to fetch specific details about media files or simply want to add a time-stamped comment to flag an issue, it happens conversationally.

## Core Capabilities

**01 — Check project status and team lists**

Retrieve comprehensive metadata for all projects in your teams, or list every collaborator involved with a given project.

**02 — Browse and detail media assets**

List folders or individual files within a project structure, and pull deep metadata about any specific asset.

**03 — Monitor feedback threads**

See all existing comments on an asset, or add new time-coded feedback directly from your agent chat.

**04 — Coordinate review links**

List and access the specific external links used for project reviews so you know who's looking at the media.

## One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/frameio** — connect your AI agent in three steps.

01 Subscribe to this MCP through Vinkius Marketplace.

02 Enter your Frame.io Personal Access Token, found in the Developer Portal.

03 Start asking your agent questions about projects or assets using any compatible AI client.

The bottom line is that once connected, you talk to your agent like it's already logged into Frame.io and pulling data for you.

## Built For

Creative Directors, Post-Production Managers, and Video Producers need this. If you spend too much time copy-pasting feedback from email or switching between five different tabs just to check project status, this is for you.

**Video Producer**

Checks if all necessary assets are available by listing folders and fetching detailed metadata before the final render.

**Creative Director**

Gets a real-time overview of who is working on which project by listing team members and checking review links for status updates across multiple departments.

**Post-Production Manager**

Audits the project history by retrieving all asset metadata and compiling comprehensive lists of time-coded comments for reporting.

## What Changes When You Connect

01 Stop switching between tabs to check on status updates. You can ask your agent for `list_projects` or `get_project_details` and get a summary immediately.

02 Never lose track of feedback again. Use the MCP to `list_asset_comments` so you see every single note, including who said it and when.

03 Build an audit trail in your chat history. You can use `add_comment` to post time-coded approval notes directly into the asset timeline as if you were using Frame.io itself.

04 Keep all context centralized by listing team members via `list_collaborators`. This single command shows exactly who is involved in which project, no guesswork needed.

05 Get deep technical data instantly. Instead of manual inspection, use `get_asset_details` to pull metadata on any file or folder right into your chat.

## Real-World Applications

**Reviewing a new campaign asset**

The Creative Director needs to know if the final video is ready for external stakeholders. They ask their agent to `list_review_links` and get an immediate list of links, instantly showing which external parties have access.

**Auditing a complex project**

The Post-Production Manager needs to compile all feedback for the client report. They ask their agent to `list_assets` and then use `get_asset_details` on key files, gathering necessary metadata and asset lists into one place.

**Updating project status quickly**

The Video Producer needs to update a client's awareness of the current status. They ask their agent for `get_project_details` on 'Summer Campaign', instantly confirming which phase it's in and who is assigned.

**Tracking feedback across multiple assets**

The Editor needs to see all comments left over the last week. They ask their agent to `list_asset_comments` for a specific video, getting a chronological log of notes from everyone.

# Patterns to Avoid

### Using email threads for feedback

✗ AVOID

The team is arguing over which revision is final. They have 15 emails referencing 'the link' and manually tracking changes in a Google Sheet.

✓ INSTEAD

Use the MCP to `list_asset_comments` or `add_comment`. This keeps all official feedback, including time-codes, inside the platform history.

### Checking project status via UI

✗ AVOID

The manager has to navigate three separate menus and click through several dropdowns just to see if a team is active or what assets exist.

✓ INSTEAD

Simply ask your agent to `list_projects` or `list_folders`. It retrieves the data for you, skipping all the clicks.

### Copying metadata into spreadsheets

✗ AVOID

The producer needs a report listing asset names and ownership. They spend an hour going through the web interface copying text fields one by one.

✓ INSTEAD

Use `list_assets` combined with `get_asset_details`. Your agent pulls all that data and gives it to you in a structured format.

## The Right Fit

Use this MCP if your workflow revolves around shared, time-sensitive media review. You need to know who is involved ( `list_collaborators` ), what the asset is made of ( `get_asset_details` ), and where feedback lives ( `add_comment` , `list_asset_comments` ). Don't use it if you just need general file storage or task ticketing; those are better handled by dedicated project management tools. However, if your goal is merely to manage billing details or list internal directories without referencing actual media assets, the MCP might be overkill. Stick to this when video collaboration and metadata are the core pain points.

## The Feedback Loop Headache

Today, reviewing a single piece of video feels like doing five different jobs. You open email for comments, jump to Slack for quick notes, check Google Drive for the latest version, and then finally go to Frame.io just to see who approved it—all with constant context switching and copy-pasting.

With this MCP, you talk to your agent instead of clicking through five different tools. You ask what changed since yesterday, and it pulls up every relevant asset detail, comment thread, or project status update in one natural response.

## Frame.io: Asset & Comment Management

You eliminate the need to manually check which folders exist, list out every collaborator on a file, or track down specific metadata points about an asset version.

Now you just ask your agent for `list_teams` and it gives you the full organizational map. It's conversationally accurate, instantly available data.

# Frame.io: 12 Tools for Media Collaboration

These tools allow you to list everything from teams and assets to project metadata and individual comments, all managed through your AI agent.

| # | TOOL | DESCRIPTION |
|---|---|---|
| 01 | `add_comment` | Posts a new, specific comment on an asset at a certain point in the video timeline. |
| 02 | `get_asset_details` | Retrieves deep metadata and technical details about any media file or folder. |
| 03 | `get_my_profile` | Gets basic information about the user who is logged into Frame.io. |
| 04 | `get_project_details` | Fetches detailed metadata and status information for a specific project. |
| 05 | `list_accounts` | Lists the billing accounts associated with your Frame.io organization. |
| 06 | `list_assets` | Retrieves a list of files or folder contents within a project library. |
| 07 | `list_collaborators` | Lists all team members and external users who have access to the current project. |
| 08 | `list_asset_comments` | Retrieves a list of existing comments made against a specific video asset. |
| 09 | `list_folders` | Lists all sub-folders contained within a given project directory. |
| 10 | `list_projects` | Retrieves a list of active projects belonging to your team or organization. |
| 11 | `list_review_links` | Accesses and lists the specific review links for monitoring external stakeholder viewing. |
| 12 | `list_teams` | Retrieves a list of all teams within your Frame.io organization structure. |

# See It in Action

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

**U** List all my projects in Frame.io team 'team_abc123'.

Retrieving projects... I found 4 active projects in your team, including 'Spring Campaign' and 'Product Launch Video'. Would you like to see the assets for any of these?

**U** Show me the last 5 comments on video asset 'vid_9876'.

Fetching comments... I found 5 recent updates for 'vid_9876', including a request to 'fix the color grade at 00:45' and a note about 'audio levels being too low'.

**U** Add a comment to 'vid_9876': 'Great work, let\'s proceed to export' at 120 seconds.

Comment added! Your feedback has been successfully posted to 'vid_9876' at the 02:00 mark. Your team will see your approval in the project timeline.

# Frequently Asked Questions

01 **How do I get an Access Token for Frame.io?**

You can generate a Personal Access Token in the Frame.io Developer Portal at https://developer.frame.io.

02 **Can I add comments with a specific timecode using the agent?**

Yes! The 'add_comment' tool accepts an optional 'timestamp' parameter in seconds, allowing you to pin your feedback to a specific moment in the video.

### 03 How do I see the assets inside a specific folder?

Use the 'list_assets' tool and provide the Folder ID as the 'parentId'. The agent will return all files and sub-folders contained within that directory.

### 04 Is it possible to list external review links?

Yes! Use the 'list_review_links' tool and provide the Project ID. The agent will retrieve all active shared links for that project.

# 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 `"frameio": { "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

# Frame.io 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 Frame.io. 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 | Frame.io MCP |
| Server ID | `019d759e-936c-7166-a747-225f143257ef` |
| 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/frameio.