MCP SERVER | NO CODE | CLOUD HOSTED

Flickr MCP

Programmatically Manage Visual Research & Metadata

Flickr MCP gives your AI client full control over public photo discovery and image data. Use it to search across billions of photos, list user albums, pull technical metadata like dimensions and source URLs, and monitor real-time visual trends directly through conversation.

A+ Quality Score 100/100

photo-sharing | image-metadata | visual-search | photography-community | media-management

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

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.

Flickr MCP

12 tools available
Cloud-hosted on Vinkius

You can connect your Flickr account to any compatible agent and treat the entire platform like a massive, searchable database for visual research. Instead of clicking through photo galleries or manually sifting through pages, you ask your agent what you need. It handles the heavy lifting. Need to find all images tagged 'Kyoto street life' from a specific group? You just ask. Want to see which tags are currently hot across the site? Your agent retrieves that list and can even pull associated photos for review. Whether you use it in Claude, Cursor, or any other MCP-compatible client, the system handles complex data retrieval—from listing public albums to getting granular details about a single image's capture date. By connecting through Vinkius, you get access to this deep visual context right alongside your other services.

Core Capabilities

01 — Search for specific visuals

Query the entire public photo archive using keywords and retrieve photos matching those criteria.

02 — Gather technical image data

Pull detailed metadata from any photo, including size, source URL, title, and description.

03 — Track visual trends

Get a list of currently popular tags and discover photos flagged as 'interesting' by the platform's algorithms.

04 — Browse user collections

List all public albums or photo sets belonging to a specific user ID for review.

05 — Explore group content

Find photos and details within designated public groups, helping identify niche visual communities.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to the Flickr MCP on Vinkius and generate your API Key from the developer portal.

02 Connect your AI client (like Claude or Cursor) using the stored credentials in your agent's settings.

03 Ask your agent a natural language question, such as 'Show me photos of Renaissance art captured last year,' and review the structured data response.

The bottom line is that you get programmatic access to billions of public images without leaving your chat interface or development environment.

Built For

This tool belongs to visual content curators, digital strategists, and developers who need large-scale image data. It's for the user tired of spending hours manually browsing photo sites just to find a reference or a trend.

Content Curator

Uses the MCP to instantly locate highly specific, relevant public images needed for mood boards and editorial research, eliminating manual searching.

Creative Strategist

Monitors real-time visual trends using `get_hot_tags` and tracks algorithmically 'interesting' photos to keep content ideas culturally current.

Developer / Data Scientist

Integrates specific functions like `get_album_photos` or `search_groups` into custom internal dashboards, pulling structured photo metadata for analysis.

What Changes When You Connect

01 Stop guessing what's trending. Use `get_hot_tags` to get a real-time list of popular tags, ensuring your content strategy always hits current cultural nerves.

02 Avoid sifting through endless galleries. Instead, ask for photos related to 'Art Deco architecture in Miami,' and let the MCP use `search_photos` to deliver immediate results with technical metadata.

03 When you need context, don't just look at a photo. Use `get_photo_info` to pull the title, description, and exact capture date for archival research or citation purposes.

04 `get_user_albums` lets you quickly list all public photosets for an account, providing a clear directory of their work without needing to visit every single profile page manually.

05 It's not just about finding pictures; it’s about structure. The MCP allows you to gather data from `search_groups` or specific user IDs and put it into a clean format your agent can process.

06 Developers gain massive efficiency by using tools like `get_album_photos` to automate the collection of visual assets, feeding them directly into custom internal tooling.

Real-World Applications

Building a Mood Board for a Client

The client needs 50 images that capture 'desert tones' and 'industrial decay.' Instead of keyword searching manually, the agent runs `search_photos` with specific filters. The resulting list gives the curator direct source URLs and dimensions, ready to be dumped into a design program.

Tracking Competitor Visual Content

A marketing team needs to know what visual themes are gaining traction among niche photography groups. They use `search_groups` first, then run `get_group_photos` on the top three results, allowing them to analyze content patterns without leaving their workflow.

Archiving User Portfolios

A developer needs a complete inventory of all visual work for a client. They use `get_user_albums` followed by iterating through each album using `get_album_photos`, building a comprehensive, structured data set.

Researching Visual Trends for an Article

A journalist needs to know what's popular right now. They ask the agent to execute `get_hot_tags`. The resulting tags guide their subsequent research, which they can then refine by asking for 'interesting photos' associated with those specific topics.

Patterns to Avoid

Assuming general web search works

✗ AVOID

Asking the agent to 'find me a picture of a dog on Instagram.' The system fails because it only has access to Flickr's specific public API endpoints.

✓ INSTEAD

To find similar content, use `search_photos` with descriptive keywords like 'Golden Retriever park portrait'. This ensures you are querying the correct, structured database.

Asking for private data

✗ AVOID

Requesting a user's full photo history or deleted photos. The MCP is restricted to publicly visible information.

✓ INSTEAD

Stick to listing public collections by running `get_user_albums` and then accessing the contents using `get_album_photos`. This stays within the scope of what the API exposes.

Copy-pasting massive photo lists

✗ AVOID

Pasting hundreds of URLs into the prompt hoping the agent processes them all. The system will hit token limits and fail to categorize or analyze the data.

✓ INSTEAD

Use `get_photo_info` for specific, targeted analysis. If you need a bulk operation, use your code client with structured loops around the `get_user_public_photos` tool.

The Right Fit

Use this MCP if your workflow requires querying massive, publicly visible visual archives and analyzing image metadata (dimensions, source URLs, titles). It's ideal for content strategists or data scientists doing visual research. Don't use it if you need access to private account details, copyrighted material outside the public domain, or real-time interactions like direct chat support. If your goal is simply browsing photos without structured data extraction, a standard web browser is better. However, if you need to programmatically list user albums using `get_user_albums` or track community interest via `get_hot_tags`, this MCP is exactly what you need.

The headache of manual visual research

Right now, finding specific visual references feels like archaeological work. You open Flickr, type in a keyword, and then you're faced with endless pages of results. You have to manually click through dozens of albums, check the metadata on each image, and copy-paste the right URLs into a spreadsheet just to build a small mood board.

With this MCP, that entire manual process vanishes. Instead of clicking links and reading captions, you simply ask your agent: 'Find me five photos showing abandoned factories in New Orleans.' The response delivers not only the images but also all the structured data points—the source URLs, dimensions, and titles—in one go.

Flickr MCP gives instant visual context

You no longer have to navigate multiple tabs or jump between a search page and an album view just to get details. The agent consolidates the information, whether it's running `get_user_albums` for a complete list of collections or using `search_groups` to pinpoint niche communities.

The difference is speed and structure. You don't collect data; you execute research queries that return actionable, structured intelligence.

Flickr MCP: 12 Available Tools for Visual Research

These twelve tools let you programmatically find, categorize, and extract structured data from Flickr's massive public image database.

#	TOOL	DESCRIPTION
01	`get_group_photos`	Retrieves all public photos contained within a designated group pool.
02	`get_hot_tags`	Returns an up-to-date list of the most popular and trending tags on Flickr right now.
03	`get_interesting_photos`	Fetches photos that have been flagged by the platform's algorithms as particularly noteworthy or interesting.
04	`get_photo_info`	Retrieves comprehensive details, including title and description, for one specific photo URL.
05	`get_album_photos`	Fetches all photos that belong to a specified public album or photoset.
06	`get_recent_photos`	Retrieves a list of the most recently uploaded public photographs across the platform.
07	`get_user_info`	Gathers general profile information about a specific Flickr user account.
08	`get_user_albums`	Lists all public photo albums belonging to a given user ID.
09	`get_user_popular_tags`	Identifies the most frequently used or popular tags associated with a particular user's content.
10	`get_user_public_photos`	Retrieves a collection of public photos uploaded by a specific user ID.
11	`search_groups`	Searches the platform's directory and returns results for designated Flickr groups.
12	`search_photos`	Performs a comprehensive search across all public photos using keywords or filters.

See It in Action

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

U Search for photos of 'Cyberpunk Architecture' on Flickr.

Retrieving photos... I've found 5 stunning images of Cyberpunk Architecture. They feature neon-lit cityscapes from Tokyo and Hong Kong. Would you like the direct source links for these photos?

U Show me what tags are trending on Flickr right now.

Fetching hot tags... Currently, trending tags include 'spring2026', 'astrophotography', and 'street-style'. Would you like to see the most 'interesting' photos associated with any of these?

U List the public albums for user '12345678@N00'.

Retrieving albums for user 12345678@N00... I've found 3 public photosets: 'Summer Trip 2025', 'Macro Nature', and 'Black & White Portraits'. Which album would you like to explore?

Frequently Asked Questions

01 **How does the Flickr MCP handle search filtering?**

The MCP supports highly specific searches using `search_photos`. You can filter by keywords and even pull metadata like dimensions, which is much more precise than a basic web search.

02 **Can I use the Flickr MCP to get private photos?**

No. This MCP only accesses public photo data. It cannot retrieve images or information from accounts that are set to private visibility.

03 **What is the difference between `get_user_public_photos` and `get_album_photos`?**

`get_user_public_photos` gets a general feed of all photos uploaded by a user. In contrast, `get_album_photos` requires you to specify an existing album ID first.

04 **Does the Flickr MCP track current trends?**

Yes. You can use the `get_hot_tags` tool to retrieve a list of currently popular tags, which helps keep your content strategy relevant and timely.

05 **Do I need developer skills to use the Flickr MCP?**

No. While it has development tools like `search_groups`, you interact with it entirely through natural language prompts in your AI client, requiring zero coding knowledge from the user.

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 `"flickr": { "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

Flickr 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 Flickr. 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	Flickr MCP
Server ID	`019dd0f2-97fe-7365-82c3-2406fa3aa623`
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/flickr.