MCP SERVER | NO CODE | CLOUD HOSTED

AppFollow MCP for AI Agents

Analyze App Store Reviews and Track Global Rankings

AppFollow gives your AI client deep visibility into app store performance. Track star ratings, monitor daily ranking changes across global charts, and analyze thousands of user reviews instantly to understand public sentiment. It's built for rapid app reputation management.

F Quality Score 14.04/100

app-reviews | sentiment-analysis | aso | customer-feedback | reputation-management | store-rankings

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

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.

AppFollow MCP

8 tools available
Cloud-hosted on Vinkius

Running an app means constantly managing its reputation. AppFollow brings powerful insights from major app stores directly to your AI agent. Instead of manually checking multiple sites or spending hours sifting through language barriers, you can ask your AI client to analyze all user feedback and performance metrics at once. You'll get instant summaries on overall sentiment, see exactly why rankings dipped last week, and compare how your app performs against competitors globally. Connecting AppFollow via Vinkius lets any compatible AI client—like Cursor or Claude—access this intelligence in natural conversation, turning raw review data into actionable product insights for PMs and marketing teams alike.

Core Capabilities

01 — Analyze User Feedback Sentiment

Generates summarized insights from user reviews, identifying common complaints, praised features, and overall emotional tone.

02 — Track Store Rankings History

Monitors your app's performance in store charts, detailing daily changes in visibility within specific categories or countries.

03 — Retrieve App Metadata and Info

Gathers foundational data about an app, including its official name and market details across various platforms.

04 — Review Specific Star Rating Distributions

Accesses the current star rating breakdown of your app to understand where users are leaving feedback (e.g., 1-star vs. 5-star).

05 — Compare App Performance Against Rivals

Compares key metrics, such as ratings and rankings, between your application and specified competing titles.

One Click on Vinkius — From Prompt to Execution

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

01 You tell your AI client what you need to know about the app—for example, 'Give me a summary of recent 1-star reviews.'

02 Your agent uses this MCP's tools to query AppFollow's database for the specific review data and ranking history.

03 The results return structured data that your AI client processes into natural language summaries, trend reports, or actionable lists.

The bottom line is you get structured app store intelligence delivered through conversational prompts, saving hours of manual research.

Built For

This MCP is for growth marketers and product managers who are tired of guessing why their app performance dipped. If your job involves reading user feedback or tracking competitive market positions, this tool saves you from sifting through thousands of mixed-language reviews manually.

Product Manager

Uses AI summaries to quickly digest hundreds of complaints and prioritize the next set of features based on real user sentiment.

Growth Marketer (ASO)

Tracks daily rankings and star ratings changes to measure the effectiveness of new keywords or marketing campaigns immediately.

Customer Support Lead

Identifies common user issues and recurring bugs by analyzing clusters of low-rated reviews across different markets.

What Changes When You Connect

01 Stop guessing about user pain points. The `get_reviews_ai_summary` tool gives you instant, digestible insights into what users are actually complaining or praising.

02 Measure your ASO efforts accurately. Use the `get_rankings` tool to track daily performance changes and prove which keywords move the needle.

03 Understand reputation over time. The `get_ratings_history` tool maps out rating trends, letting you see if past fixes actually improved user satisfaction.

04 Save hours on competitive analysis. You can compare your app's performance directly against rivals using metrics gathered from `get_reviews_summary`.

05 Quickly assess the market. The `get_app_info` tool retrieves core metadata, giving you a baseline understanding of any competitor or target app.

06 Pinpoint specific bugs. By listing individual reviews with `list_reviews`, your AI agent can quickly identify recurring feature requests or critical bugs mentioned by users.

Real-World Applications

Investigating a sudden drop in visibility

A growth marketer notices their app dropped from #3 to #15 in the 'Productivity' category last week. They prompt the agent using `get_rankings` and then use `get_reviews_ai_summary` to check if the dip correlates with a sudden influx of negative reviews after a recent update.

Prioritizing product features

A Product Manager needs input for the next sprint. They ask their agent to analyze all user feedback using `get_reviews_ai_summary` and group complaints into top three themes, allowing them to prioritize development efforts efficiently.

Launching in a new international market

A team expanding globally wants to know the initial reception. They use `list_reviews` for multiple countries and then run `get_ratings` to gauge immediate star rating health before committing resources.

Benchmarking against a competitor's success

A marketing team wants to know why a rival app is succeeding. They use `get_app_info` and then run a comparative analysis with the agent, using data from both apps to identify best practices.

Patterns to Avoid

Reading thousands of raw reviews

✗ AVOID

A team tries to manually copy-paste 500 random reviews into a spreadsheet and ask their agent to 'tell us what's wrong.' This is slow, inaccurate, and misses the big picture.

✓ INSTEAD

Instead, use `get_reviews_ai_summary`. This tool processes thousands of reviews instantly and gives you actionable summaries, grouping complaints by category for immediate results.

Focusing only on average ratings

✗ AVOID

You see your app has a 4.2-star rating and assume everything is fine. But the problem might be concentrated in the low star reviews that aren't factored into the average.

✓ INSTEAD

Always use `get_ratings` to check the distribution, not just the mean score. This reveals if you have a large number of 1- or 2-star complaints hiding under an acceptable average rating.

Ignoring historical context

✗ AVOID

You see your app is ranked well today and assume it will stay that way forever, ignoring seasonal dips or competitive launches.

✓ INSTEAD

Track performance over time. Use `get_rankings` and `get_ratings_history` together to establish a baseline of normal performance variation, giving you context for current metrics.

The Right Fit

Use this MCP if your primary need is rapid intelligence on public sentiment and market positioning. You absolutely must know what users are saying *and* how those comments correlate with measurable store charts. For instance, if a PR campaign dropped the ratings by half a star, you need to immediately cross-reference that drop using `get_rankings` against the feedback generated by

`list_reviews`. Don't use this MCP if you only need basic app metadata; then, just querying the app ID is enough. If your goal is complex competitor modeling across multiple industries, consider a broader market intelligence tool instead of focusing solely on app stores.

AppFollow MCP for AI Agents: Diagnosing App Store Review Sentiment

Today, analyzing user feedback is a massive manual task. You have to log into the Apple App Store, then Google Play, and maybe even specialized forums. You copy dozens of reviews—some in Spanish, some in German—and paste them into a spreadsheet just to manually count how many people mention 'slow' or 'login bug.' It's tedious, slow, and you always miss nuance.

With the AppFollow MCP, your AI client handles it all. You simply ask for an AI summary of user feedback across multiple stores. The agent runs `get_reviews_ai_summary` and spits out a clean report: 'Top 3 Complaints: Slow Loading (25%), Login Bugs (18%), Missing Dark Mode (10%).' You get immediate, prioritized action items.

AppFollow MCP for AI Agents: Tracking App Store Rankings Performance

Historically, tracking rankings meant logging into a dashboard and watching charts. If your visibility dipped, you had to guess why—was it competition? Was it an algorithm change? You spent time compiling reports that were always slightly out of date.

Now, the AppFollow MCP allows your agent to run `get_rankings` and track those changes in real time relative to historical data. If a dip occurs, you immediately have context on whether it's a sudden drop or part of a normal seasonal fluctuation. You get clear, actionable performance metrics.

AppFollow: 8 Tools for Analyzing App Store Reviews and Rankings

Use these tools to get specific data points, whether you need a current star rating count or historical ranking trends across global markets.

#	TOOL	DESCRIPTION
01	`get_account_check`	Verifies that your AppFollow account is properly connected for usage.
02	`get_app_info`	Retrieves basic, foundational details about a specific app from the store.
03	`get_rankings`	Tracks your app's current position in various relevant store charts.
04	`get_ratings_history`	Provides historical data on star ratings, showing trends over a selected time period.
05	`get_ratings`	Gathers the current distribution of star ratings for your app (e.g., how many 5-star vs. 3-star reviews exist).
06	`get_reviews_ai_summary`	Generates a concise, AI-powered summary of the most recent user feedback.
07	`get_reviews_summary`	Provides an overall synthesis of reviews, including average rating and key themes.
08	`list_reviews`	Lists individual app reviews for a specific product in any major store market.

See It in Action

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

U What are the main complaints about my app since last month?

App Review Summary: Last 30 Days

Based on over 950 reviews, the overall sentiment is positive (4.1/5 stars), but three major themes emerged:

- 🦎 **Performance:** Frequent reports of slow loading time after updates.
- 🎨 **UI/UX:** Multiple requests for a dark mode option and better widget customization.
- ⚙️ **Features:** Users are asking to integrate with calendar services directly.

Action: Prioritize addressing the performance issues first.

U How did my ranking change in 'Productivity' last week?

Ranking Trend: Productivity Category (US)

The app started at #12 on Monday and ended at #9 today. This is an increase of 3 spots.

- **Monday:** #12
- **Tuesday:** #11 (+1)
- **Wednesday:** #10 (+1)
- **Thursday:** #9 (+1)

This upward trend suggests recent marketing efforts are paying off.

U Give me a summary of 1-star reviews for the Canadian market.

Canada Review Digest (1-Star Reviews)

Reviewers consistently point to two issues:

1. **Localization:** The app fails to correctly display localized currency symbols in settings.
2. **Offline Mode:** It completely loses synchronization when the internet drops, making it unusable while traveling.

Frequently Asked Questions

01 **How does AppFollow help me understand what users really think about my app?**

AppFollow analyzes user feedback and gives you summarized insights on sentiment. Instead of reading hundreds of reviews, your AI agent tells you the top issues—like 'slow loading' or 'missing dark mode'—and how many times they were mentioned.

02 **Can I use AppFollow MCP to track my app's performance against competitors?**

Yes, it lets you compare your app directly with rivals. You can pull data on ratings and rankings for multiple apps simultaneously, helping you figure out where your competitive edge is.

03 **What kind of rating history data does AppFollow provide?**

It gives you historical star rating trends over time. This means you can see if a feature fix or marketing push actually caused a measurable, positive shift in overall user satisfaction.

04 **Is the AppFollow MCP only for US-based app stores?**

No, it covers major global app stores and multiple countries. You can list and summarize reviews from different languages and regional markets to get a worldwide view of your reputation.

05 **How do I use AppFollow MCP to find out why my rankings dropped?**

You can track the rank changes using `get_rankings` and then cross-reference that date range with the reviews. The AI agent will help correlate the drop in visibility with spikes in negative user feedback.

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 `"appfollow": { "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

AppFollow 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 AppFollow. 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	AppFollow MCP
Server ID	`019d7550-1859-712d-b600-364d89d63c3a`
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/appfollow.