MCP SERVER | NO CODE | CLOUD HOSTED

Appbot MCP for AI Agents

Analyzing App Reviews and Tracking User Feedback from iOS and Android

Appbot gives you deep insights into user feedback for your mobile app. It lets your AI client analyze reviews—whether from iOS or Android—to track sentiment, identify bug reports, and uncover key topics instantly. You can filter massive amounts of text by star rating, country, or specific keywords without leaving your chat interface.

A+ Quality Score 100/100

sentiment-analysis | app-reviews | user-feedback | customer-insights | mobile-apps | rating-tracking

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

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.

Appbot MCP

10 tools available
Cloud-hosted on Vinkius

Appbot lets you turn thousands of app store reviews into actionable product data. Instead of manually sifting through a spreadsheet of complaints, you talk to your agent and it handles the heavy lifting. You can programmatically pull in raw review text, analyze the overall tone—is it positive, negative, or mixed?—and categorize every piece of feedback by common themes. It even tracks changes across different app versions so you know exactly when a new release caused problems. Because Vinkius hosts Appbot alongside 4,000 other MCPs, you connect your AI client once and get access to this review analysis tool plus hundreds of others, making it the single source for all your operational data.

Core Capabilities

01 — List all tracked applications

See every app managed within Appbot by running list_apps.

02 — Retrieve specific user reviews

Get the full text and details for any single review using get_review_details. It's great for deep dives into specific complaints or praise.

03 — Filter by custom topics

Pull together all reviews associated with a pre-defined group of keywords or themes via get_reviews_by_custom_topic.

04 — Find reviews across different countries

Understand your global audience by filtering reviews using the list_countries tool to narrow down results by region.

05 — Analyze all standard and custom themes

List common topics identified in user feedback, whether they're built-in standards or custom themes defined in your dashboard (list_custom_topics).

06 — Narrow down reviews by keywords and sentiment

Run a comprehensive search for reviews using list_reviews, letting you filter by star rating, specific words, or the overall sentiment.

One Click on Vinkius — From Prompt to Execution

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

01 First, connect Appbot to your AI client and provide the necessary API credentials.

02 Next, ask your agent a question. For example: 'Show me all the negative reviews for the last two versions in the UK.'

03 Your agent runs the relevant tools (like list_reviews or get_review_details) and presents you with filtered data, sentiment summaries, and topic breakdowns.

The bottom line is, your AI client handles the API calls. You just ask questions about your user feedback, and it returns structured insights immediately.

Built For

Product Managers who get buried in raw data; Support Leads drowning in repetitive complaints; App Developers needing to prioritize bug fixes from real users.

Product Manager

You use this MCP to evaluate the reception of new features. You'll ask your agent to pull reviews by specific topics or star ratings to identify pain points before they become major issues.

Customer Support Lead

When a pattern emerges, you instruct your agent to monitor sentiment trends for that issue across all platforms. You can quickly gauge if an emerging bug is localized or global using list_languages.

App Developer

After deploying a new build, you use this MCP to check technical feedback and bug reports from the app store reviews specifically for that version (list_versions). This helps prioritize your sprint backlog.

What Changes When You Connect

01 Pinpoint the exact cause of frustration. You don't just see 'negative'; you use list_reviews to filter by keywords, instantly finding out if users hate a specific button or feature.

02 Manage global feedback in one place. Using list_countries lets your agent group reviews from different regions, letting you compare how 'Performance' is discussed in Germany versus Canada.

03 Go deeper than simple star ratings. By using get_reviews_by_custom_topic, you can pull together all comments related to 'Login Issues,' regardless of whether the user gave 1 or 5 stars.

04 Evaluate releases instantly. When a new version drops, use list_versions and list_reviews to track feedback specifically for that build, letting developers know exactly what needs fixing right now.

05 Consolidate all data sources. Because Vinkius hosts Appbot with thousands of other MCPs, you keep your workflow centralized—you don't need a separate dashboard just for reviews.

Real-World Applications

A new feature is failing in the market.

The Product Manager asks: 'What are the top three pain points regarding the checkout flow?' The agent uses list_reviews and get_reviews_by_custom_topic to return a prioritized list of complaints, showing that 70% of issues relate to payment gateway compatibility.

Support team needs to know if an issue is spreading.

A Support Lead asks: 'Check all reviews from France and Italy mentioning 'sync error' over the last week.' The agent uses list_languages, list_reviews, and get_review_details to provide a geo-specific report, indicating the problem started after a specific app version was pushed.

Developers need help prioritizing bug fixes.

The App Developer asks: 'List all reviews for our Android app version 3.2 that mention 'crashes' or 'slow.' What is the sentiment?' The agent uses list_versions and list_reviews to deliver a quantitative count of technical bugs versus general complaints.

Understanding seasonal shifts in user focus.

The Product Manager asks: 'What are the most common topics mentioned in reviews globally during Q4?' The agent uses list_topics and list_countries to analyze trends, showing that 'holiday gift exchange' became a dominant theme months ahead of usual reporting.

Patterns to Avoid

Searching only by keywords

✗ AVOID

The user asks the agent: 'Show me reviews mentioning login.' This returns hundreds of results, mixing actual bugs with simple feedback.

✓ INSTEAD

Instead, ask for a filtered report: 'Using list_reviews, show me reviews from the last month that mention 'login' AND have a star rating less than 2 stars.' This narrows the focus to actionable complaints.

Ignoring app versions

✗ AVOID

Running general sentiment analysis without context. You see negative feedback, but don't know if it's from last year or yesterday.

✓ INSTEAD

Always use list_versions first. Then ask: 'List reviews for version 4.5 that have a negative sentiment.' This pins the problem to a specific code release.

Over-relying on general topics

✗ AVOID

Seeing Appbot suggests 'Usability' is a topic, but not knowing *why* users find it hard.

✓ INSTEAD

Use get_reviews_by_custom_topic combined with list_reviews. For example: 'List reviews for the payment flow using the custom topic 'Checkout Pain Points.' This gives you raw quotes and specific feedback.

The Right Fit

Use Appbot if your problem is too much unstructured text; specifically, if you need to find patterns in thousands of app store reviews. Don't use this if your data already lives cleanly in a structured database like SQL or Google Sheets—in that case, standard ETL tools are better. If you only care about the raw

sentiment score without context, other general NLP services might suffice. However, Appbot is superior because it provides the crucial ability to combine filtering criteria (like star rating and country) with topic identification, for instance, asking your agent to find 'Negative reviews for Version 3.1 in Japan.' This level of contextual specificity requires this MCP.

Appbot: Turning App Reviews into Actionable Product Insights

Right now, product teams spend hours jumping between the Apple and Google developer consoles. They export CSVs of reviews, then they run them through separate sentiment analysis tools. It's a tedious process of copy-pasting data into spreadsheets just to find out if users are consistently complaining about the same bug or feature.

With this MCP, you ask your agent one question: 'What are my top 5 issues?' The system runs list_reviews and gets_review_details behind the scenes. You get a structured report that ranks the problems by frequency AND severity, letting you skip the spreadsheet mess entirely.

Appbot: Tracking Global User Feedback Across App Platforms

Without this MCP, if your product launches in a new country, you have to manually set up tracking and wait for data ingestion. You can't easily compare how the core feature is perceived by users in Brazil versus Australia.

Appbot solves that complexity. By using list_countries and getting_review_details, you instantly pull together geographically segmented reports. It means your product strategy is always informed by actual global user sentiment, not just aggregated averages.

Appbot: 10 Tools for Analyzing User Feedback & Sentiment

Use these tools to gather comprehensive data on app performance, filter reviews by region or star rating, and categorize user feedback into specific topics.

#	TOOL	DESCRIPTION
01	`get_account_info`	Pulls Appbot account details and confirms your connection status with the service.
02	`get_review_details`	Provides complete, granular information for a single, specific user review.
03	`get_reviews_by_custom_topic`	Collects all reviews that fall under a custom topic you've set up in your Appbot account.
04	`list_apps`	Displays every application name and ID currently tracked by the team within Appbot.
05	`list_countries`	Lists all geographical regions available for filtering review data to understand local market feedback.
06	`list_custom_topics`	Retrieves the list of user-defined thematic categories you've set up in the Appbot dashboard.
07	`list_languages`	Shows all languages supported by Appbot for accurate sentiment analysis and filtering.
08	`list_reviews`	Lists reviews for a specific app, allowing optional filters by keywords, star rating, or general sentiment to narrow results quickly.
09	`list_topics`	Retrieves the list of standard themes identified within app reviews using Appbot's built-in AI analysis.
10	`list_versions`	Detects and lists all specific application versions that have received user feedback in the reviews.

See It in Action

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

U What were the main complaints about the app last week?

Weekly Sentiment Report: Last 7 Days

• **Overall Trend:** Slightly more negative (Avg. Star Rating: 3.1/5)
• **Top Complaint Category:** Performance Issues (28% of negative reviews)
Key Bug:* App crashes during payment processing.
Affected Version:* v4.1
• **Trending Topic:** 'Syncing' is mentioned 30% more often than the week prior, suggesting a recent backend change broke something.

U Show me the most common topics for my US audience.

Top Topics in Reviews (USA)

Based on reviews from the last 30 days, here are the top three themes:

1. **Performance:** Mentioned frequently, often linked to slow loading times.
2. **User Interface:** Users report difficulty finding settings menus.
3. **Payment Flow:** Specific complaints about required multi-factor authentication steps.

Action: Focus on simplifying the user flow for payments in v4.2.

U Compare feedback from Canada versus the UK.

Comparative Review Analysis

FEATURE	CANADA (CA)	UNITED KINGDOM (UK)
Sentiment	Mostly neutral, focusing on features.	Negative, centered on stability.
Top Issue	Missing integration with local banking apps.	App crashes when switching between tabs.

Insight: The UK audience is focused on immediate reliability; the Canadian audience wants more advanced feature parity.

Frequently Asked Questions

01 **How does Appbot help me find specific bugs in user reviews?**

Appbot lets you run focused searches using list_reviews. You can combine filters like 'star rating less than 3' and a keyword (like 'crash') to pull only the bug reports. This cuts through generic complaints so developers know exactly what needs fixing first.

02 **Can Appbot analyze reviews from different countries at once?**

Yes, it can. You use list_countries and list_reviews to group feedback by region. This is critical for seeing if a feature works correctly across your global user base or if the issue is localized to one market.

03 **Is Appbot better than just reading our internal support tickets?**

Appbot adds vital context that support tickets lack. It gives you the public perception—what users *think* about your product, not just what they told a human agent. You can identify emerging problems before they hit your ticket queue.

04 **How do I find out if a new app version caused complaints?**

You use list_versions to pinpoint the exact build number and then run reviews against it. This immediate feedback loop means you don't have to wait weeks for manual reports; you know what broke right after deployment.

05 **Does Appbot just track positive comments, or can it find problems?**

It finds everything. You control the sentiment filter when running reviews, so you can specifically pull negative or mixed feedback. This ensures your focus stays on addressing pain points, not celebrating praise.

06 **Can Appbot compare themes across multiple apps I run?**

Yes. By listing all apps first and then running topic analysis against each one, you can see if a common theme (like 'login') is impacting your whole product suite or just one specific app.

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 `"appbot": { "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

Appbot 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 Appbot. 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	Appbot MCP
Server ID	`019d754f-b289-725c-901c-756d689c3c74`
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/appbot.