MCP SERVER | NO CODE | CLOUD HOSTED

Foursquare MCP

Find and analyze locations globally.

Foursquare gives you deep control over location data, letting your AI agent find, verify, and analyze any point of interest (POI) in the world. Search millions of venues using natural language; check operating hours, audit photo content, or map density within specific geographic areas like polygons or radii. It's geospatial intelligence built for conversation.

A+ Quality Score 100/100

geospatial-data | poi-search | location-intelligence | venue-data | routing | mapping

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

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.

Foursquare MCP

10 tools available
Cloud-hosted on Vinkius

Connecting Foursquare lets your AI client handle complex location discovery without manual searches. Instead of clicking through dozens of tabs to verify a store's details or check surrounding businesses, you just ask. You can run deep queries, finding what exists near a specific GPS pin, or mapping density across an entire neighborhood using custom boundaries. It retrieves rich data like operating hours and user ratings instantly. When you combine this with the Vinkius catalog, your agent gets access to one of the industry's deepest sets of location tools. You can even automatically predict search terms as you type, making any conversation about a physical place feel precise and immediate.

Core Capabilities

01 — Search for places anywhere

Find points of interest across the globe by executing general searches or limited queries within defined geographic areas.

02 — Define custom search boundaries

Run complex searches that map to specific shapes, like polygons, or measure distance from a central point using radii.

03 — Retrieve detailed venue profiles

Get structural data on a location, including its operating hours, official ratings, and full taxonomy classification.

04 — Analyze user-generated content

Pull in raw text reviews and dynamic image URLs to audit the quality and social buzz of specific venues.

05 — Predict location search terms

Get fast, accurate typeaheads by querying partial letters, helping your agent quickly narrow down intent.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to the Foursquare MCP and enter your unique API key in the Vinkius developer console.

02 Connect this MCP to any compatible client (like Claude or Cursor).

03 Instruct your agent with natural language prompts, and it executes complex location searches and data retrieval using the tools.

The bottom line is you get deep access to Foursquare's full suite of place discovery data directly through conversation.

Built For

This MCP is essential for anyone whose job relies on physical location data. Think urban planners, logistics managers, and app developers who need precise, real-time information about the world's venues without relying on manual web scraping.

Real Estate Analyst

You use this to audit commercial density or check venue quality across a large neighborhood polygon before advising a client.

Logistics Planner

You verify the operational hours and nearest POIs for multiple warehouses or delivery points in a specific radius.

App Developer

You test location-based search parameters, ensuring your app can handle ambiguous inputs by forcing single venue matches.

What Changes When You Connect

01 Check operating hours instantly. Instead of visiting a website to find out if 'The Corner Bistro' is open right now, your agent uses `get_place_details` to confirm its status immediately.

02 Map neighborhood density. Use `search_within_polygon` or `search_nearby_venues` to audit how many similar venues exist in a specific commercial zone for market analysis.

03 Understand user buzz. Running `get_place_tips` pulls in raw visitor reviews, helping you gauge the public sentiment and quality of any location beyond just the star rating.

04 Validate data precision. If an address is ambiguous, use `match_venue_exactly`. This forces Foursquare to narrow down results so your agent can confidently deliver one single answer.

05 Build smart autocomplete. The `autocomplete_venues` tool predicts what you're looking for as you type, making complex location searches feel natural and fast.

Real-World Applications

Verifying a market area for expansion

A city analyst needs to know if there are enough competing coffee shops in the new district. They use `search_within_polygon` to draw the exact boundaries and then run general searches across all found venues to assess POI density.

Checking event logistics

A planner needs to confirm if a pop-up shop will be accessible. They prompt for details near a target pin, using `search_nearby_venues` and then checking the specific hours with `get_place_details`.

Debugging location search forms

A developer needs to ensure their app doesn't return multiple results for one vague input. They use `match_venue_exactly` to validate that the system forces a single, correct venue node.

Patterns to Avoid

Treating location data like static text.

✗ AVOID

Trying to find out if 'The Coffee Spot' is open on Sunday using only basic search queries. These often return general results without current status or hours.

✓ INSTEAD

You must use `get_place_details` for specific venues. This tool retrieves the necessary structural data, giving you real-time operating hours instead of just listing the name.

Overlooking geographic boundaries.

✗ AVOID

Assuming a nearby location is actually within your target zone when it's slightly outside the intended area.

✓ INSTEAD

Don't trust general searches. You need to use `search_within_polygon` or `search_within_radius` to enforce hard, defined physical boundaries around your search.

Ignoring user context in research.

✗ AVOID

Just listing a venue's address without knowing if customers liked it. This gives you zero insight into local appeal or quality.

✓ INSTEAD

Always use `get_place_tips` to pull in raw, human-written reviews and recommendations alongside the official data for a complete picture.

The Right Fit

Use this MCP if your problem is fundamentally about physical location: 'Is X near Y?' or 'What's happening inside this specific boundary?'. You need to know if a place exists, what its hours are, or how many similar places crowd an area. Don't use it if you simply need general knowledge (like historical facts) or complex calculations that don't involve geography. If your goal is to analyze the internal structure of data types—for example, classifying everything in the world using Foursquare's system—then `list_venue_categories` gives you that master list. But if you just need to search for a product category (like 'coffee'), use general searches; this MCP adds the crucial layer of geospatial precision.

The headache of manually checking multiple local listings.

Right now, finding out everything about a spot—its hours, if it's busy, or what's around it—is a click-and-copy nightmare. You jump from Google Maps to the business site for hours, then maybe check Yelp for reviews, and finally use another tool just to map how many other competitors are nearby in that same zip code. It’s tedious, fragmented, and takes forever.

With this MCP, your agent handles it all in a single conversation. You ask: 'What's the vibe at The Corner Bistro, and is it open right now?' And you get back structured details—the hours, the rating, and even recent user tips—all synthesized instantly.

Foursquare MCP gives you total control over location data.

The process of defining a search area used to involve complex manual drawing or guessing radii. You'd spend time verifying if the boundary was right and if your query covered every corner of the target zone.

Now, you specify exact boundaries—whether it’s a custom polygon shape or a precise radius measurement around a pin—and the MCP runs the search immediately. It delivers clean, actionable data for everything within that defined space.

Foursquare: 10 Tools for Geospatial Intelligence

These tools let your agent perform precise location searches, extract detailed venue profiles, map density patterns, and analyze user-generated content across the globe.

#	TOOL	DESCRIPTION
01	autocomplete_venues	Provides fast suggestions for venues when a user types partial names or location keywords.
02	list_venue_categories	Exports the entire structured classification tree (taxonomy) used by Foursquare to categorize businesses.
03	get_place_photos	Retrieves direct media URLs for images associated with a specific location.
04	get_place_details	Gathers comprehensive information on a place, including its official hours and detailed schema properties.
05	get_place_tips	Collects raw user reviews and tips left by visitors about the venue.
06	match_venue_exactly	Runs a validation check to ensure that an ambiguous search query returns only one specific, confident location result.
07	search_nearby_venues	Finds venues located within a specified distance (radius) from a given GPS coordinate.
08	search_within_polygon	Searches for all points of interest contained entirely within a custom-drawn, multi-sided geographic boundary.
09	search_within_radius	Performs rich searches by defining a precise distance scope around a target coordinate.
10	search_places	Executes broad, general queries across the entire Foursquare POI graph to identify initial search results.

See It in Action

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

U Find coffee shops near '40.71, -74.00'

Searching places... I found 10 coffee shops nearby. Top results include 'Blue Bottle Coffee' (0.2 miles away) and 'Everyman Espresso'. Would you like the detailed ratings or hours for 'Blue Bottle'?

U What are the opening hours for 'Central Park Zoo'?

Retrieving details for Central Park Zoo... They are open today from 10:00 AM to 5:00 PM. It is currently rated 4.5/5 by over 2,000 visitors. Would you like to see some recent photos of the zoo?

U Show me user tips for 'The Met Museum'

Retrieving tips... Visitors recommend: 'Go early to avoid crowds', 'Don't miss the Egyptian wing', and 'The rooftop view is incredible'. I found 50+ detailed tips for your visit.

Frequently Asked Questions

01 **How do I find places near a specific coordinate using Foursquare MCP?**

You use `search_nearby_venues` or `search_within_radius`. You just provide the coordinates and the desired search scope (the radius), and the agent finds all relevant POIs.

02 **Can I check if a business is open using Foursquare MCP?**

Yes, run `get_place_details` on the venue. This tool pulls in structured information that includes current operational hours and status for verification.

03 **What does 'polygon' mean when I use Foursquare MCP?**

A polygon is a custom, multi-sided shape used to define an exact geographic area. Using `search_within_polygon` lets you analyze only what falls inside your drawn boundaries.

04 **Is there a way to predict what I want to search for with Foursquare MCP?**

Yes, use the `autocomplete_venues` tool. It provides fast typeaheads by querying partial letters and predicting user intent as you talk to your agent.

05 **How do I get all the possible categories for a venue search using Foursquare MCP?**

You use `list_venue_categories`. This tool exports the entire official taxonomy, letting you see every classification rule available in the system.

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 `"foursquare": { "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

Foursquare 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 Foursquare. 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	Foursquare MCP
Server ID	`019d759e-7755-7000-9dae-20c7096db0e9`
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/foursquare.