MCP SERVER | NO CODE | CLOUD HOSTED

NVIDIA AI MCP

Accelerate Reasoning and Model Inference

NVIDIA AI MCP connects your agent directly to industry-leading, GPU-accelerated foundation models. It lets you chat with large language models like Llama or Mistral, generate code from simple prompts, convert natural questions into SQL queries, and create vector embeddings for advanced search—all without managing complex infrastructure.

A+ Quality Score 100/100

llm | gpu-acceleration | embeddings | model-inference | natural-language-processing | code-generation

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/nvidia-ai

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.

NVIDIA AI MCP

9 tools available
Cloud-hosted on Vinkius

This MCP gives your agent direct access to the power of NVIDIA's API Catalog. You don't have to worry about GPU hardware; you just use what you need. Need your AI client to write Python code? Use the `generate_code` tool. Want to know if a piece of text is positive or negative? Run sentiment analysis right away. You can even feed natural language questions into the system and convert them into functional SQL queries using `text_to_sql`. Beyond basic chat, you can generate vector embeddings for advanced search, condense massive reports with summarization, or translate content across dozens of languages. When you connect this MCP via Vinkius, your agent gets instant access to all these capabilities from a single point, making complex AI tasks simple commands.

Core Capabilities

01 — Advanced Reasoning

Ask deep questions and receive answers generated by powerful reasoning models.

02 — Chat with Large Language Models

Engage in conversations using top-tier foundation models like Llama 3.1 or Mistral.

03 — Vector Embedding Creation

Turn any block of text into a numerical vector for use in search, clustering, and retrieval systems.

04 — Code Generation

Write functional code snippets—like Python or JavaScript—by giving the agent a simple description of what you want.

05 — Natural Language Data Querying

Convert human-readable questions into precise SQL queries that can interact with databases.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to the NVIDIA AI MCP and enter your personal API key from build.nvidia.com.

02 Select this MCP within your preferred client, like Cursor or Claude.

03 Your agent can now call tools directly—for example, running `chat_completion` to chat with Llama 3.1.

The bottom line is that you connect the API key once and gain access to dozens of GPU-backed models through your AI client's tool library.

Built For

This MCP is for developers who need robust, high-performance AI capabilities without managing the underlying infrastructure. It helps data scientists move from concept to deployment faster and lets business analysts query complex systems using everyday language.

Data Scientist

Uses `get_embeddings` to index large datasets for vector search or runs NLP tasks like sentiment analysis at scale.

Software Developer

Uses the `generate_code` tool to quickly prototype API endpoints and write boilerplate code within their IDE.

Business Analyst

Employs `text_to_sql` to ask questions about company metrics in plain English, getting a ready-to-use database query back.

What Changes When You Connect

01 Generate working code on demand. Instead of leaving the chat window to use a separate tool, your agent can call `generate_code` right away, writing full snippets like FastAPI APIs based only on your prompt.

02 Go from question to query instantly. Stop drafting SQL queries manually for every data request. Use `text_to_sql` to convert natural language into database code with zero friction.

03 Handle massive amounts of text efficiently. Need a quick digest of a 50-page report? Run the `summarize_text` tool and get the core findings without reading through filler paragraphs.

04 Power up your search functionality. Instead of keyword matching, you can use `get_embeddings` to create dense vector representations of documents for true semantic retrieval.

05 Stay in one place. By connecting this MCP via Vinkius, your agent gets access to everything—from chatting with Llama 3.1 using `chat_completion` to analyzing sentiment—without switching services.

Real-World Applications

Analyzing Customer Feedback at Scale

A data scientist receives thousands of customer reviews and needs to know the overall mood. They ask their agent to run `analyze_sentiment` on all the text, grouping results by 'negative' sentiment so they can immediately flag critical issues for the product team.

Building a Knowledge Retrieval System

A developer needs an internal wiki search engine. They first run `get_embeddings` on all existing documents, then use those vectors to power a semantic search that finds relevant context when responding to user queries.

Translating and Summarizing Global Content

A marketing analyst receives a long white paper written in German. They first run `translate_text` into English, then feed the result into `summarize_text` so they can create quick, accurate summaries for local press releases.

Interacting with Internal Databases

A business analyst needs Q3 sales data but doesn't know the underlying schema. They simply ask their agent, 'What were the top selling products in Q3?' and use `text_to_sql` to generate the exact query needed for the BI tool.

Patterns to Avoid

Over-relying on basic chat

✗ AVOID

Asking a simple, general LLM model (like one used only for `chat_completion`) to write complex API code or structure SQL queries.

✓ INSTEAD

Don't just chat with the model. Use specific tools like `generate_code` when you need functional code, or use `text_to_sql` when you are talking about databases. These dedicated tools force structured output.

Mixing up embedding and text generation

✗ AVOID

Trying to search a knowledge base using only keywords after running the standard chat tool.

✓ INSTEAD

For true semantic search, always run `get_embeddings` on both your query and your documents. This creates vectors that allow your agent to find meaning, not just matching words.

Assuming language capability

✗ AVOID

Asking the LLM to translate a document without confirming its multilingual support.

✓ INSTEAD

Always use the dedicated `translate_text` tool. It guarantees neural translation across dozens of languages, which is far more reliable than general chat completions.

The Right Fit

Use this MCP if your workflow requires deep model interaction, especially when you need to move beyond simple text generation. You need it when your process involves querying structured data (use `text_to_sql`), converting unstructured data into searchable formats (`get_embeddings`), or generating runnable code (`generate_code`). If your only requirement is a basic conversation—just asking general questions—you might get by with a simpler, general-purpose chat tool. But if you need to interact with databases or build production-ready applications, this MCP is essential because it provides the highly specialized tools that turn pure language models into actionable agents. Don't use this just for simple translation; use `translate_text` when you require high fidelity across many languages.

Dealing with data silos and context switching

Today, if your agent needs to answer a question about sales figures, you have to copy the query into a database tool. If it needs to write code based on that finding, you paste the result into an IDE and then ask another service for review. It's constant copying, pasting, and jumping between three or four different interfaces.

With this MCP, your agent manages the entire loop. You simply tell your client what you need—like asking 'What was the Q2 revenue growth?' The system handles calling `text_to_sql` to get the query, running it against the data source, and then summarizing the result for you in a single chat thread.

Getting structured code from unstructured ideas with generate_code

Before this MCP, writing even small functions required opening an IDE, setting up file structures, and manually referencing API documentation to ensure the syntax was perfect. It felt like starting a new project every time.

Now, you just describe the function—'Write a Python class that connects to a Postgres database.' The `generate_code` tool returns a fully formed, ready-to-use code block instantly. You get working code, not suggestions.

NVIDIA AI: 9 Tools Available

These tools let your agent perform specific tasks like running sentiment analysis, chatting with large language models, and generating code using GPU acceleration.

#	TOOL	DESCRIPTION
01	`ask_question`	Asks a question using a powerful reasoning model with optional context for better answers.
02	`chat_completion`	Chats with an NVIDIA AI model (Llama, Mistral, etc.) by specifying the desired model name and conversation history.
03	`generate_code`	Creates code from a natural language prompt when you specify a programming language.
04	`get_embeddings`	Generates vector embeddings for any given text using the specified NVIDIA model.
05	`list_models`	Provides a list of all AI models currently available through the entire NVIDIA API Catalog.
06	`text_to_sql`	Converts natural language questions into executable SQL queries for database interaction.
07	`analyze_sentiment`	Determines the emotional tone (positive, negative, neutral) of a provided piece of text.
08	`summarize_text`	Condenses long documents or articles into short, concise summaries while retaining key information.
09	`translate_text`	Translates text accurately between dozens of supported languages.

See It in Action

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

U Generate Python code for a REST API with FastAPI.

Generated code:
```
from fastapi import FastAPI
app = FastAPI()

@app.get('/items')
async def get_items():
return {'items': []}
```

U Translate 'Hello, how are you?' to Japanese.

こんにちは、お元気ですか？ (Konnichiwa, ogenki desu ka?)

U Summarize: The quarterly report shows revenue grew 15% YoY...

Q3 revenue increased 15% year-over-year, driven by strong demand in AI and cloud services.

Frequently Asked Questions

01 **How does the NVIDIA AI MCP help with embedding vectors?**

The `get_embeddings` tool converts any text into a numerical vector using the specified model. This is crucial for advanced search, allowing your agent to find conceptual matches instead of relying only on exact keywords.

02 **Can I use chat_completion with different models?**

Yes, you specify which AI model—like Mistral or Llama 3.1—you want to talk to directly within the `chat_completion` tool call, giving you control over performance and style.

03 **What is text_to_sql used for?**

The `text_to_sql` tool translates human language questions into accurate SQL queries. This lets your agent query databases without needing to know the database schema or write complex syntax.

04 **Is summarize_text good enough for legal documents?**

It's excellent for condensing long texts, but remember it is a summary tool. For highly sensitive legal review, you should always pair `summarize_text` with detailed context provided through the chat completions.

05 **Does NVIDIA AI MCP support multiple programming languages?**

The `generate_code` tool allows you to specify various languages. You just need to tell your agent what language you want, and it writes the code in that syntax.

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 `"nvidia-ai": { "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

NVIDIA AI 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 NVIDIA AI. 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	NVIDIA AI MCP
Server ID	`019d75e0-d789-73e2-834a-6c437b160898`
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/nvidia-ai.