MCP SERVER | NO CODE | CLOUD HOSTED

Leonardo.ai MCP

Control every step of your AI art pipeline.

Leonardo.ai (Generative AI & Models) lets you build complex creative pipelines directly into your agent. Generate high-fidelity images using precise prompts or guide the process by uploading reference photos. You can discover, manage, and audit custom models and generation history all without leaving your chat client.

A+ Quality Score 100/100

generative-ai | image-generation | model-management | visual-assets | prompt-engineering | content-production

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

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.

Leonardo.ai (Generative AI & Models) MCP

10 tools available
Cloud-hosted on Vinkius

This MCP connects your generative image workflow to Leonardo.ai, giving you full control over professional visual assets through simple conversation. Instead of navigating a complex web dashboard, you tell your AI agent exactly what you want—whether that's generating an image from scratch or expanding a piece using a reference photo.

Your agent handles the whole process: it can list available models, check which custom styles are loaded onto your account, and initiate asynchronous generation requests. If you need to guide the style, you upload a source image and generate variations based on its structure. You'll also get real-time visibility into your usage by monitoring token allocations. Because Vinkius hosts this MCP within its catalog, you can connect all your specialized AI services—image creation, data querying, workflow automation—from one place, making your entire creative stack accessible to your agent.

Core Capabilities

01 — Generate Images

Send a text prompt and get an image generation request ID that you can then poll until the final result is ready.

02 — Track Generation Status

Check if a previous image generation request is still running or if it has successfully completed.

03 — Manage Models and Styles

List all public platform models, discover your specific fine-tuned custom styles, or retrieve detailed parameters for any model.

04 — Create Visual Variations

Upload an initial image to generate structural extensions or context variations that expand the original piece while maintaining its style.

05 — Audit Image History and Usage

List all past user generations, retrieve links, track the prompts used, and monitor your current token usage against account limits.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP and provide your Leonardo.ai API key within your agent's settings.

02 Ask your AI client to perform a specific action, like 'Generate an image of a robot in the style of Kino XL.'

03 Your agent calls the necessary functions, giving you status updates until the high-resolution images are ready.

The bottom line is, you use plain language prompts to control complex, multi-step generative tasks without ever leaving your chat interface.

Built For

This connector is built for creative teams and high-volume digital artists. It solves the pain of switching between a web dashboard, an API playground, and a project management tool just to create one asset.

Digital Designer

Needs to quickly iterate on visual concepts by generating variations or extending initial sketches using specific models.

Creative Director

Requires oversight of the team's creative output, managing generation history and monitoring collective token usage for budget control.

AI Content Producer

Must automate high-fidelity asset production, running batches of themed imagery across multiple marketing campaigns consistently.

What Changes When You Connect

01 Stop guessing which model to use. Use the `list_platform_models` and `list_custom_models` tools to see exactly what styles are available before you start generating.

02 When an image isn't quite right, don't scrap it. Upload a source picture using `upload_init_image` and generate variations based on its composition instead of starting over.

03 Keep track of your budget and usage in real time. The `get_user` tool gives you live metrics so you know exactly how many tokens are left for the week.

04 Need to build a specific style? Use the `get_model` function to pull model details, ensuring your agent uses the exact parameters required for consistency across projects.

05 Managing history is easy. You can list all past assets with `list_user_generations`, or if you need to clean up records, use `delete_generation`.

Real-World Applications

Maintaining Brand Consistency Across Assets

A marketing team needs 50 variations of a product shot. Instead of manually running the same prompt and model fifty times, they ask their agent to use `upload_init_image` with the master photo, then loop through `create_variation` until all required angles are covered.

Auditing Project Costs

A creative director needs to know if a team member is running up costs. They ask their agent to use `list_user_generations` and then `get_user` to cross-reference the history with current token usage, ensuring compliance.

Developing New Visual Styles

A concept artist wants to see if a new model works. They use `list_platform_models` first, select 'Phoenix,' and then run a test generation via `generate_image` with a complex prompt to validate the style.

Troubleshooting Failed Generations

An asset fails to load. Instead of restarting, the agent uses `get_generation` with the original ID to check if the failure was due to model parameters or network issues, saving time.

Patterns to Avoid

Treating it like a simple image host

✗ AVOID

Just typing 'make me an astronaut' and expecting the best result immediately. This ignores your account limits or preferred style.

✓ INSTEAD

Always start by asking the agent to check `list_custom_models` first, then use that model UUID in the prompt when you call `generate_image`. This guarantees brand consistency.

Ignoring image context

✗ AVOID

Generating a new image but needing it to match the perspective or style of an existing one. The result looks disconnected.

✓ INSTEAD

Before generating, use `upload_init_image` with your reference photo. Then ask the agent to execute a variation using that secure URL.

Over-generating and forgetting costs

✗ AVOID

Running dozens of test generations without realizing how quickly tokens deplete, leading to unexpected billing alerts.

✓ INSTEAD

Check your operational budget first. Always use `get_user` before starting a large batch job so you know exactly how many generation cycles are left.

The Right Fit

Use this MCP if the core of your workflow involves iterative, high-fidelity image creation that requires model selection, style guidance, and historical auditing. You need control over the *process*, not just the output file.

Don't use it if you only need to generate a single, simple image once in a while—a standard web interface is fine for that. Also, don't use it if your primary goal is text-only content generation; those are better suited for dedicated LLM MCPs. You must be comfortable with the idea of structured data retrieval (like checking metrics or listing models) because this tool gives you technical control over every aspect of image production.

The biggest time sink in creative work isn't making images; it's managing them.

Currently, creating a campaign's worth of assets means toggling between three places: the main Leonardo dashboard to pick a model, a separate history tab to check if an image was already made, and then maybe opening another tool just to see your token balance. You spend more time clicking through tabs and copying UUIDs than you do actually creating art.

With this MCP, that entire process collapses into one conversation with your agent. You tell it: 'Create five variations of the robot image using Model X.' The agent handles the model selection, the generation request, and even tells you if you're running low on tokens. You get back immediate, actionable results.

Accessing Model Metadata with get_model

Before this MCP, figuring out which model was best for a specific style meant trial and error—generating dozens of images just to compare results. You were blind guessing in the dark.

Now, you can ask your agent to run `get_model` on 'Kino XL' or any other style. You get the full technical parameters immediately. That capability alone cuts out hours of pointless testing and lets you focus only on what works.

Leonardo.ai (Generative AI & Models) 10 Tools

These tools give you direct programmatic control over every aspect of the Leonardo.ai platform, from initial prompt generation to final asset management.

#	TOOL	DESCRIPTION
01	`generate_image`	Generates a new image based on your text prompt and returns an ID to track the output status.
02	`get_generation`	Checks the current status of any active or completed image generation request using its unique ID.
03	`list_user_generations`	Retrieves a list of all recent images and their associated metadata created by your account.
04	`list_platform_models`	Lists all global, publicly available models hosted on the Leonardo.ai platform.
05	`list_custom_models`	Retrieves a list of private, fine-tuned models that you've trained and uploaded to your account.
06	`get_model`	Fetches detailed information and specific parameters for any given Leonardo.ai model by name or ID.
07	`delete_generation`	Permanently removes a specific image generation log entry from your history record.
08	`get_user`	Retrieves current account metrics, including remaining tokens and usage statistics for billing oversight.
09	`create_variation`	Generates an unzoomed extension of a previously created image to expand its context while keeping the style consistent.
10	`upload_init_image`	Acquires a secure link needed to upload a source image for advanced image-to-image guidance.

See It in Action

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

U Generate a futuristic cityscape at sunset using the Phoenix model

Generating image... I've triggered a request using the Phoenix model (UUID: 123-abc). Your generation ID is 'gen-98765'. I'll monitor the status and provide the high-resolution links as soon as they are ready.

U List my last 3 image generations

I've retrieved your recent generations: 1) 'Cyberpunk Forest' (gen-101), 2) 'Minimalist Logo' (gen-102), and 3) 'Abstract Portrait' (gen-103). Would you like me to fetch the direct URLs for any of these?

U Check my current token balance and account limits

Retrieving metrics for user 'alex_designer'... You currently have 1,240 tokens remaining for today. Your daily limit is 2,000. You have performed 45 generations so far in this billing cycle.

Frequently Asked Questions

01 **How do I start generating an image using Leonardo.ai (Generative AI & Models) MCP?**

You initiate the process by asking your agent to generate an image and providing a detailed text prompt, like 'A futuristic city at sunset.' The tool will then return a Generation ID for tracking.

02 **Can I use Leonardo.ai (Generative AI & Models) MCP to guide my images?**

Yes, you can. By using `upload_init_image`, you provide your agent with a source photo, and it will give you the necessary secure link for guided generation or variations.

03 **What is the difference between list_platform_models and list_custom_models in Leonardo.ai (Generative AI & Models) MCP?**

Platform models are the public, global styles available to everyone. Custom models are the specific, private fine-tuned styles that you or your team trained on your own account.

04 **Is there a way to check my usage limits with Leonardo.ai (Generative AI & Models) MCP?**

Absolutely. Use the `get_user` tool, and it will retrieve your active account metrics, including tokens remaining for today's generation budget.

05 **How do I access my older images with Leonardo.ai (Generative AI & Models) MCP?**

You simply ask the agent to run `list_user_generations`. It will pull up a list of your recent work, along with their direct URLs and original prompts.

06 **Does Leonardo.ai (Generative AI & Models) MCP let me delete old history?**

Yes. If you need to clean up records or remove sensitive data, the `delete_generation` tool lets you explicitly erase a specific generation log entry.

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 `"leonardoai-generative-ai-models": { "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

Leonardo.ai (Generative AI & Models) 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 Leonardo.ai (Generative AI & Models). 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	Leonardo.ai (Generative AI & Models) MCP
Server ID	`019d75c6-a5ce-7245-b87f-46ecc98d6765`
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/leonardoai-generative-ai-models.