MCP SERVER | NO CODE | CLOUD HOSTED

DevDocs MCP for AI Agents

Search and retrieve technical manuals for programming libraries

DevDocs provides conversational access to a global index of programming documentation, letting your AI agent search and read technical manuals from hundreds of libraries. Stop switching tabs or searching Stack Overflow; this MCP pulls specific API definitions, framework guides, and SDK details directly into your workflow.

A+ Quality Score 100/100

api-documentation | technical-reference | sdk-documentation | search-indexing | programming-languages | developer-resources

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

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.

DevDocs MCP

3 tools available
Cloud-hosted on Vinkius

Need to reference an obscure function signature or check the precise syntax for a niche AWS command? This connector gives your AI agent direct access to DevDocs.io's vast technical documentation index. Instead of manually navigating dozens of websites, you talk to your agent and it pulls the exact code examples and definitions you need. It handles everything from listing available frameworks like Vue 3 or React, to finding a specific manual page path within an SDK. You can then read the full content—cleanly formatted Markdown—without leaving your IDE or chat window. With Vinkius hosting this MCP, you connect once and gain control over one of the most important research tools for any developer.

This lets you ground your coding tasks in real-time technical context, making it a huge help whether you're building an agent or just trying to remember how `useState` works.

Core Capabilities

01 — List available programming libraries and frameworks

Your AI client retrieves the full list of supported languages and SDKs, like AWS, Rust, and Vue 3.

02 — Locate exact documentation page paths

The agent searches a specific library's index to find the precise manual page path for any component or class name.

03 — Read and format full technical documentation content

It fetches and translates native HTML documentation into clean, human-readable Markdown text for easy consumption.

One Click on Vinkius — From Prompt to Execution

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

01 Subscribe to this MCP. No credentials are required because it uses the public DevDocs.io registry.

02 Tell your AI client what you need, whether it's listing all available languages or searching for a specific API component.

03 The system returns cleanly formatted Markdown content detailing the documentation, including usage syntax and code examples.

The bottom line is, you get to treat technical research like chatting with an expert who has read every manual ever written.

Built For

This MCP targets professional software developers, AI engineers building agents, and technical writers. If your job involves constantly looking up API definitions or cross-referencing framework standards across multiple languages, this is for you. You'll stop context switching between browser tabs to find a single piece of syntax.

Software Developer

You use it to quickly look up specific API references or SDK definitions without leaving your IDE.

AI Engineer

You ground your coding agents in accurate, version-specific technical documentation to stop them from generating outdated code.

Technical Writer

You cross-reference API signatures and verify documentation blocks across multiple frameworks while drafting guides.

What Changes When You Connect

01 Instead of juggling browser tabs, you can list all supported languages (like AWS or Vue 3) using the `list_libraries` tool to see exactly what documentation is available.

02 You stop guessing API syntax. Use `search_docs` to find the exact manual page path for a component, saving minutes on every single lookup.

03 The `read_page` tool pulls complex technical guides and formats them into clean Markdown. You get usable content immediately, no messy HTML parsing required.

04 It provides version-specific context. When writing code, your agent can pull documentation chunks that are guaranteed to match the library's current definition.

05 This MCP makes it possible for AI agents to truly operate as technical researchers, improving the quality and accuracy of generated code.

Real-World Applications

Need to find a specific command syntax across multiple cloud providers

An agent uses `list_libraries` to confirm 'aws' is supported. It then uses `search_docs` for the required resource (e.g., S3) and finally uses `read_page` to pull the exact, up-to-date command syntax in Markdown format.

Writing a technical guide that requires cross-referencing multiple APIs

The writer uses this MCP to verify API signatures across two different libraries. The agent retrieves documentation chunks using `read_page` and confirms the correct versioning boundaries for both services.

Building a new feature using an unfamiliar framework

The AI engineer first queries the list of supported frameworks via `list_libraries`. They then use `search_docs` to find the specific hooks required for their component, getting reliable code context instantly.

Patterns to Avoid

Pasting raw HTML documentation

✗ AVOID

Manually copying a code snippet from an API website often results in messy, unreadable HTML that breaks formatting when pasted into notes or another document.

✓ INSTEAD

Use the `read_page` tool. It pulls the full content and translates it directly into clean Markdown, making it ready to paste anywhere without losing structure or style.

Searching documentation via general Google queries

✗ AVOID

A basic search query like 'how to use X in Y library' returns dozens of links, forcing the user to click through multiple pages just to find the right manual section.

✓ INSTEAD

Use `search_docs` first. It searches the official index and directs you straight to the exact page path for what you need.

Assuming documentation is universally available

✗ AVOID

Starting development on a new language or SDK without knowing if reliable, version-specific guides exist.

✓ INSTEAD

Always start with `list_libraries`. This verifies that the entire framework or toolset you need is within the DevDocs global registry before you invest time in coding.

The Right Fit

Use this MCP if your primary bottleneck is finding precise, reliable technical context. If you constantly find yourself searching for API signatures, checking library versions, or needing clean documentation formats across different languages, this tool solves that pain point directly. Don't use it if you just need general conceptual knowledge—for that, a large language model works fine. But if you need to confirm the usage of `useState` in React, or find the correct syntax for an AWS S3 command, you must use DevDocs MCP. Relying on memory or generalized AI responses will only lead to outdated code; this tool gives you ground truth.

DevDocs: Solving API Documentation Research Pain with DevDocs

Today, finding accurate technical documentation is a huge time sink. You're bouncing between Stack Overflow threads, vendor websites, and outdated README files. You copy-paste code snippets that sometimes fail because they use deprecated syntax or assume you know the exact required library version. It’s context switching hell.

With this MCP, your agent handles all that friction. Instead of leaving your workflow to navigate multiple tabs, you simply ask for a specific definition. The system pulls valid, version-specific documentation chunks directly into Markdown format. You get reliable technical truth instantly.

DevDocs: Improving Code Context and Library Discovery with DevDocs

The manual process of checking which frameworks are compatible or finding the correct starting point for a new project is slow. You might spend hours just listing what tools exist before you can even write the first line of code.

Now, you use `list_libraries` to see everything available in one shot. This immediate oversight allows your agent to ground its suggestions in reality, meaning every piece of suggested code or API call is vetted against a global registry.

DevDocs: 3 Tools for Technical Reference & Library Indexing

These tools allow you to list supported languages, search specific documentation indexes, and pull formatted technical pages into your workflow.

#	TOOL	DESCRIPTION
01	`list_libraries`	Lists all supported programming languages, frameworks, and SDKs available in the DevDocs global registry.
02	`search_docs`	Searches a specific documentation library index to find the exact manual page path for components or classes.
03	`read_page`	Reads the full content of a specified documentation page and returns it as cleanly formatted Markdown text.

See It in Action

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

U I need to compare the usage syntax for copying files between AWS and Docker.

Here's a quick comparison based on their official documentation:

- **AWS CLI (S3):**
  `aws s3 cp file.txt s3://my-bucket/`
  (Requires authentication token.)

- **Docker:**
  The Docker engine uses mounting volumes for local access, not a direct copy command.

For detailed usage and examples, check the specific documentation pages.

U What are all the major programming languages supported here?

I found hundreds of libraries available. The most commonly referenced include:

- **Backend:** Node.js (Express), Python (Flask),
- **Frontend:** React, Vue 3, NextJS,
- **Infrastructure:** AWS SDK, Docker CLI.

Which one should we look at first?

U What is the proper usage of 'debounce' in a JavaScript event handler?

Function: `debounce(func, delay)`

This function limits how often an action runs. It's crucial for optimizing input fields and preventing API spam.

Syntax Example:

```javascript
const debouncedSearch = debounce((query) => {
fetchSearchResults(query);
}, 300);
```

Frequently Asked Questions

01 **Can DevDocs help me find the right syntax for a specific API call?**

Yes. You can ask your agent to search the documentation index and retrieve the exact manual page path. It pulls structured information, so you get accurate code examples ready to use.

02 **What if I'm using an older version of a framework? Can DevDocs handle that?**

The MCP can pull documentation for various versions and SDKs. You specify the required context, and it retrieves the corresponding guides, ensuring you don't use deprecated methods.

03 **How do I know what languages or frameworks are covered by DevDocs?**

You can ask DevDocs to list all supported libraries. This gives you a comprehensive overview of every language and SDK available in the global registry, so you know where to start your research.

04 **Does using DevDocs mean I don't need to look at official documentation sites?**

No, but it dramatically changes *how* you access them. Instead of manually clicking through multiple websites, the MCP brings the clean, formatted content directly into your conversation for immediate use.

05 **Is DevDocs good for AI agents that need code context?**

It's excellent. By connecting this MCP, you give your agent real-time access to validated technical documentation. This helps the agent generate much higher quality and more accurate code.

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 `"devdocs": { "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

DevDocs 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 DevDocs. 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	DevDocs MCP
Server ID	`019d7584-ab5b-7345-9839-830893efdfc6`
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/devdocs.