

The Context Layer · Following

★ Member-only story

Apple Dev Docs Unlocked: The One Free Tool Every Swift Dev Will Soon Swear By

6 min read · Sep 15, 2025



Jannis



Following ▾



Listen



Share



More

Stop getting AI “blank pages” — see how this audacious project put Apple’s locked-down documentation back in play for LLMs, Copilot, and real-world builders.

If you’ve ever watched your favorite code assistant or LLM freeze up when you asked for a Swift or UIKit snippet, you’re not alone.

Apple’s official documentation may be beautiful for humans — if you’re logged in, running JavaScript, and clicking around manually.

For AI models, repo-readers, or Copilot-style agents?

It’s basically sealed. I wanted to figure out if anyone had solved this modern walled-garden problem for machine learning, developers, and the new school of “MCP” (Model Context Protocol) tools.



Hi, this is Jannis.

Let me introduce you to sosumi.ai — a tool I’ve just gone end-to-end through, and I think you’ll understand my enthusiasm once you see the problem and the solution. Apple’s developer docs are rendered with JavaScript, which means when most AI tools, LLMs, or GitHub Copilot try to ingest or answer questions about iOS, Swift, or UIKit, they read... nothing. Just a placeholder telling you to turn on JavaScript. That’s the reason you get those awkward syntax suggestions or see tools like Claude or ChatGPT suddenly freeze or hallucinate Swift method signatures.

So what does sosumi.ai do differently?

The main trick: just drop any Apple dev docs URL into sosumi.ai (replace the domain), and you get a pure, AI-readable Markdown version instead of JavaScript-fogged nothingness. Swift, UIKit, Xcode, Core Data — it all just “works,” for both LLMs and humans who like clean docs. No crawling, no scraping, no rule-breaking: it converts a single page, on-demand, to Markdown, referencing Apple’s original JSON behind the scenes.

Even more interesting, sosumi.ai is built with the Model Context Protocol (MCP) in mind. That means it doesn’t just spit out static docs — it exposes developer documentation as structured Markdown, and even enables “tools” for querying and

fetching docs right inside systems like Copilot for Xcode, Claude Desktop, Cursor, or any other MCP-capable agent. There's a dead-simple HTTP API (replace the URL domain) and a robust MCP endpoint (plug into your agent or SaaS).

To use sosumi.ai, simply take any official Apple Developer documentation URL and replace `developer.apple.com` with `sosumi.ai`. For example, if you wanted to access the documentation for Swift's Array type, you'd turn the original link

`https://developer.apple.com/documentation/swift/array` into

`https://sosumi.ai/documentation/swift/array`.

That's it! You instantly get a clean, Markdown-formatted page that LLMs, plug-ins, or AI-powered tools can actually read and use.

If you're building an app, an MCP agent, or integrating documentation lookup into dev tools—whether for Swift, SwiftUI, UIKit, or advanced frameworks like CoreData, this is your way forward.

Try searching `https://sosumi.ai/documentation/uikit` or plug the API endpoint `https://sosumi.ai/mcp` directly into tools like Copilot for Xcode or Claude Desktop to enable instant, up-to-date Apple docs help in your workflow.

Why is this actually a big deal?

If you're building an AI assistant, Copilot extension, or any AI-powered SaaS tooling for macOS or iOS engineers, you hit a big wall: **you can't reliably offer accurate help, search, or code completion for Apple platforms, because the official docs are locked to bots and models.** Attempting to crawl or snapshot the docs in bulk? That crosses ethical and legal boundaries and is resource-heavy.

Sosumi sidesteps this: each page conversion is transient, doesn't mass-download, and complies with Apple's TOS.

There is an MCP for that, too

Integration for MCP is as simple as changing a config to point to its endpoint — no hacky scraping or local archive, just official docs turned useful for you.

How does this stack up against the use of Context7?

When comparing sosumi.ai to Context7 MCP for our use case, the main distinction is focus and coverage: sosumi.ai specifically unlocks the official Apple Developer Documentation for LLMs and MCP clients, turning previously inaccessible web-based docs into AI-readable Markdown, with both human-friendly URLs and a robust MCP API.

This means it's currently the only plug-and-play solution for instant, frictionless Apple docs access in agents, VS Code, Claude Desktop, Copilot for Xcode, and more — critical for anyone building tools or workflows tightly coupled to the Apple ecosystem.

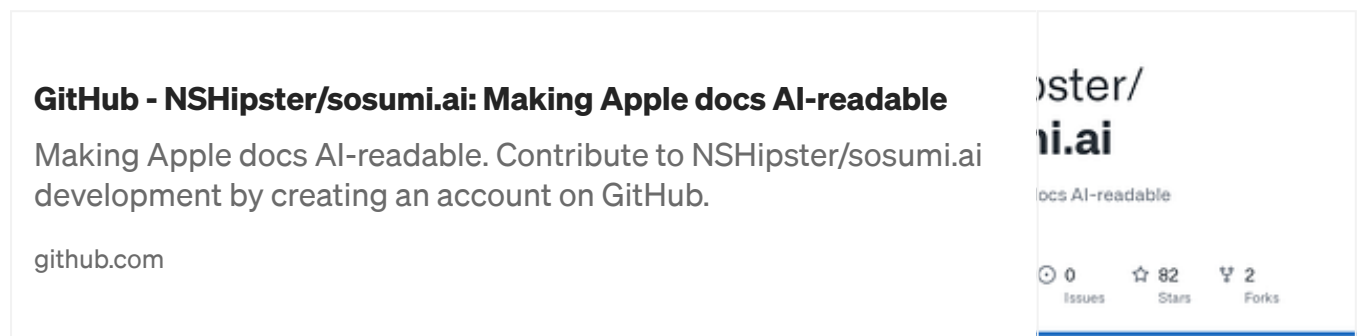
Context7 MCP, on the other hand, is broader in scope but less tailored for Apple specifics: it pulls live, version-accurate documentation and code examples from an

expanding index of open source and major third-party libraries — think React, Next.js, FastAPI, Supabase, and more — delivering that context directly into your AI agent prompts with the `use context7` phrase. It shines for web, backend, and library-centric dev work, ensuring your code generation is never out of date or train-bound. However, it isn't focused directly on Apple's JS-rendered docs, and relies on community-maintained feeds for library coverage—so for Swift/Objective-C/iOS/Mac-specific APIs, it's not as comprehensive or up-to-date as sosumi.ai.

For anyone building LLM-driven workflows, code assistants, or SaaS MVPs with deep ties to Apple's developer stack, sosumi.ai fills the documentation gap that broader tools like Context7 MCP don't fully address. If your use case expands to cutting-edge, cross-stack dev (React, Python, NPM, cloud APIs), Context7 MCP complements sosumi, but for Apple-ecosystem completeness and the best “just show me the docs that work” experience, sosumi.ai currently stands alone.

<https://sosumi.ai/>

Sosumi.ai is the first Apple documentation renderer built for the age of LLMs and MCP. It removes the “blank page” problem for AI tools, solves legal and compliance headaches, and is, in my experience, the fastest way to put Apple's official knowledge back to work in the workflows and assistants of the future.



If you're interested in strengthening your AI, dev tools, or SaaS MVP for Apple's platforms — or just want to see tech that solves a real pain — you owe it to yourself to check it out.

If you found this article helpful, A few claps 🙌, a highlight 🖋️, or a comment 💬 really helps.

If you hold that 🙌 button down something magically will happen, Try it!

Don't forget to follow me to stay updated on my latest posts. Together, we can continue to explore fascinating topics and expand our knowledge.

Thank you for your time and engagement!

. . .

Stay Ahead with The Context Layer

Your MCP Resource Hub — Get the latest Model Context Protocol news, tutorials, and practical advice delivered straight to your feed.

What You'll Get:

- **Breaking MCP News** — First to know about updates and developments
- **Step-by-Step Tutorials** — From beginner tips to advanced strategies
- **Real-World Applications** — See how MCP boosts productivity and automates workflows
- **Community Insights** — Learn from fellow developers and AI enthusiasts

Whether you're just discovering MCP or building advanced solutions, we've got you covered.

 [Follow The Context Layer on Medium](#) 

Join other developers staying current with MCP innovations.

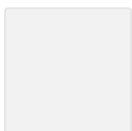
Swift

Swiftui

Swift Programming

Developer Tools

Vibe Coding



Following

Published in The Context Layer

814 followers · Last published Oct 21, 2025

A journal of real-world experiments, tools, and thinking around Anthropic’s Model Context Protocol (MCP)—now established as an open standard for large language model (LLM) context integration.



Following

Written by Jannis

2.9K followers · 5 following

Product Owner in global telecom, lifelong tech tinkerer, and Mac user. Sharing hands-on hacks, real stories, and the tools that make work (and life) smarter.



No responses yet



Bgerby

What are your thoughts?

More from Jannis and The Context Layer

 Jannis 

MS Office Without MS Office—The Free Web App That Does It All

Bring Word, Excel, and PowerPoint to your browser —no subscriptions, no sign-ins, no strings attached.

✦ Oct 30  13





 In The Context Layer by Jannis 

Stop Writing Claude Code Configurations From Scratch—This Template Library Changes Everything

If you are spending too much time figuring out Claude Code configurations, there's a comprehensive resource that can eliminate this setup...

★ Aug 3  3



 In The Context Layer by Jannis 

System Prompts Are Dead Weight: Why Parlant's Contextual Guidelines Are What Modern Agents...

The end of LLM prompt bloat — how Parlant enables aligned AI agents with modular, code-first control you can deploy in minutes.

★ Oct 10



I Built a Claude Skill in Under 30 Minutes And It Immediately Elevated My Workflow

From zero to skill in 30 minutes: the easiest way to extend Claude with your own workflow.


★ Oct 18



See all from Jannis

See all from The Context Layer

Recommended from Medium

 In The Context Layer by Jannis 

Everything Swift Developers Need, To Use MCP

Into Swift and MCP? A Swift developer just solved one of our biggest problems, and most of us don't even know it exists yet.



Aug 2

6



In Mac O'Clock by Nikhil Vemu 

iOS 26.1 Is a Massive Update—15+ New Features & Changes

Many of your iOS 26 annoyances are fixed now



5d ago

1



In AI Software Engineer by Joe Njenga

Anthropic Just Solved AI Agent Bloat—150K Tokens Down to 2K (Code Execution With MCP)

Anthropic just released smartest way to build scalable AI agents, cutting token use by 98%, shift from tool calling to MCP code execution

★ 3d ago 23



Alain Airom (Ayrom)

Second Experience with Apple Containers!

Another step with Apple containers!

Oct 17 1




The iPhone Update You MUST Avoid: Apple's Huge New Warning for Millions

If you have an iPhone, you're likely running iOS 26. But be warned. There's a dangerous hidden setting buried in your phone that you must...

★ Oct 25 💬 13



 Jonathan Woolley

Apple Just Killed the Cold Call

So, Apple just killed the cold call.... no really i think they have.

Oct 21 💬 32



See more recommendations