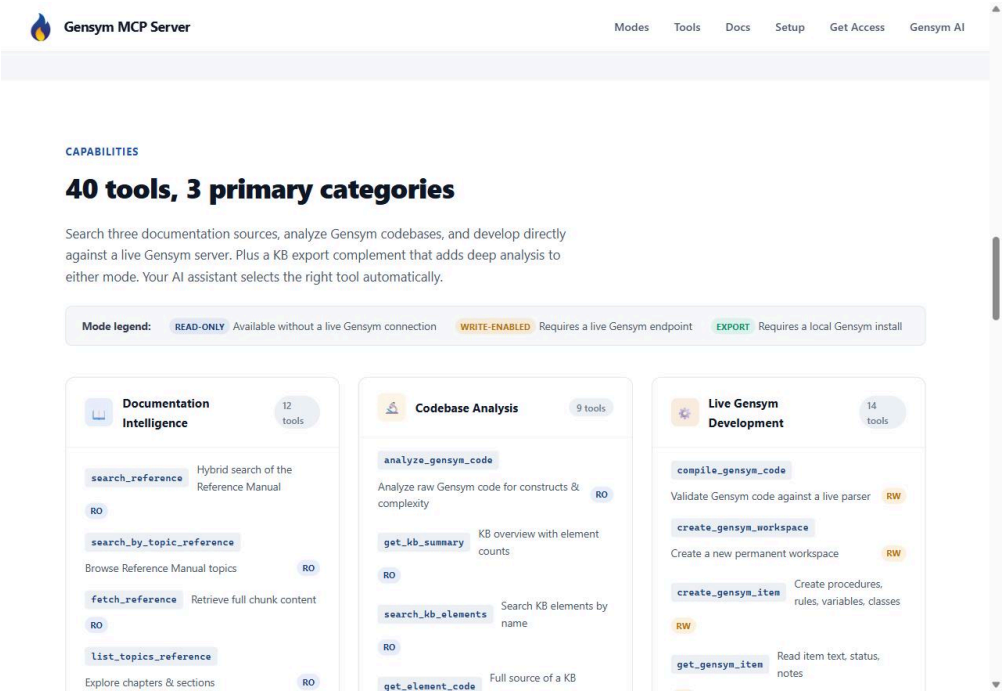
Overview

Gensym G3 AI is an MCP-based integration layer that turns any MCP-compatible AI assistant, such as **Cursor, Claude Code, Claude Desktop, or ChatGPT,** into a Gensym developer. These assistants run on a range of models, from small or local ones to frontier models, and **the 40 Gensym G3 AI tools work with all of them,** so you can match the model to your task, budget, and privacy needs. Documentation search runs on IgniteTech's hosted corpus, while codebase analysis and live development run on your own machine, so there's no server infrastructure to stand up on your side.

What's Included

- **Documentation intelligence:** Hybrid search across the Gensym Reference Manual, Tutorials, and System Procedures, making answers cited and grounded.
- **Codebase Analysis:** Summarize KBs, detect stale code, flag performance anti-patterns, and redner dependency diagrams
- **Live development:** Compile code against a live parser, create and update procedures and rules, save & merge knowledge bases, and read live runtime state.



System Requirements

- **Node.js 18+:** the MCP proxy runs via npx.
- **An MCP-compatible IDE:** Cursor, Claude Code, Claude Desktop, or ChatGPT.
- **An IgniteTech-issued API key** (header X-API-KEY).
- Optional: a local Gensym installation if you want write-enabled mode and/or local KB export.

© Ignite Enterprise Software Solutions, Inc. | All Rights Reserved

Setting Up Gensym G3 AI

1. **Request your API key.** Email success@ignitetech.ai or contact your IgniteTech Customer Success Manager. Keys are issued per organization for tracking and security.

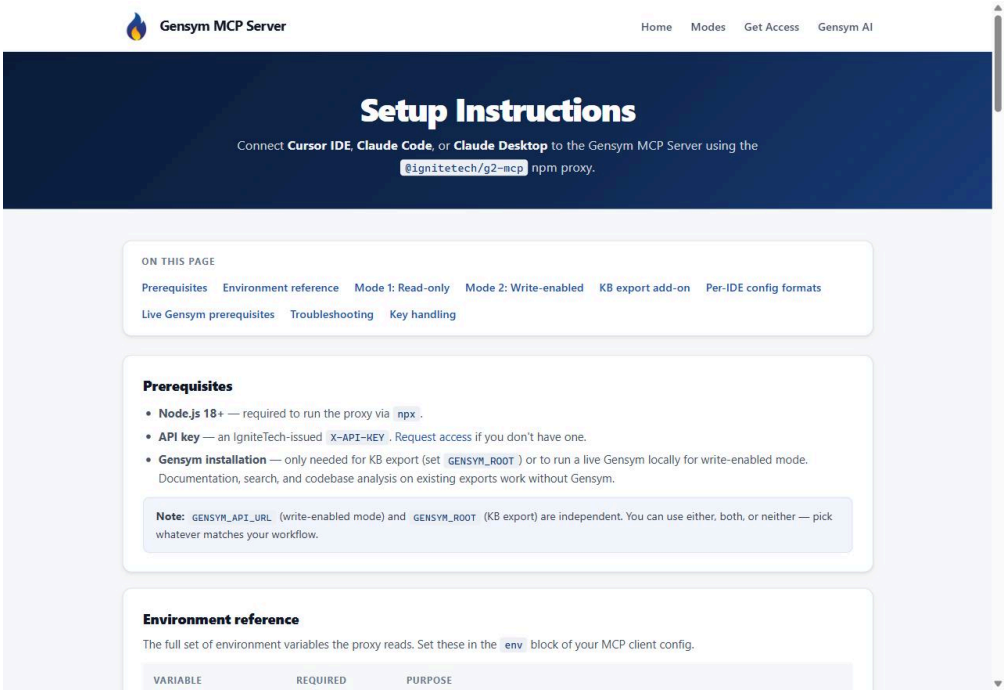
2. **Add the config snippet to your IDE**. Open your MCP config file (.cursor/mcp.json for Cursor, the Claude Desktop config for Claude Desktop, or the equivalent for Claude Code) and paste the read-only snippet below. Replace <YOUR_API_KEY> with the key you received in step 1.

```
{
  "mcpServers": {
    "gensym-mcp": {
      "command": "npx",
      "args": [
        "-y", "@ignitetech-ps-support-triage-queue/g2-mcp@latest",
        "https://g3mcp.ignitetech.ai/mcp/",
        "--header", "X-API-KEY:${GENSYM_MCP_API_KEY}"
      ],
      "env": {
        "GENSYM_MCP_API_KEY": "<YOUR_API_KEY>",
        "GENSYM_ROOT": "/path/to/your/g2/install"
      }
    }
  }
}
```

3. **Reload your IDE.** The proxy validates your key on startup; if it's valid, all read-only tools become available automatically. If not, you get a clear error message.

4. *Optional:* add GENSYM_API_URL to the env block to enable write-enabled mode against a Gensym instance you've started yourself (typically http://localhost:8082). If you set GENSYM_ROOT instead, the AI can spawn the API server for you with start_gensym_api_server, and GENSYM_API_URL is set automatically.

Full per-IDE configs (Cursor, Claude Code, Claude Desktop), environment-variable reference, KB export add-on, and troubleshooting are at **g3mcp.ignitetech.ai/setup**



© Ignite Enterprise Software Solutions, Inc. | All Rights Reserved

Two Modes at a Glance

Mode	Setup	What the AI can do
Read-only	API Key only	Search docs, analyze prior KB exports, generate Gensym code snippers that you copy-paste into your KB
Write-enabled	API key + GENSYM_API_URL	Compile, create, update, save and merge directly against alive Gensym instance on your machine, plus everything in read-only

Heads up: write-enabled mode lets the AI modify your knowledge base. Use a non-production KB for experimentation, or stay in read-only mode if you're not ready for the AI to make changes.

Privacy: Pair the tools with a local model and your data stays put. Write tools, such as compile, create, update, save, merge, run locally via npx against your own Gensym instance, so your KB and code never leave your machine. With a local model, inference stays local too, nothing goes to a third-party provider. Documentation search queries IgniteTech's hosted corpus.

What You Can Ask the AI

- "*How do I connect to a SQL database in Gensym?*" → grounded answer with citations from the Reference Manual and System Procedures.
- "*Write me a procedure to add two numbers.*" → Gensym code generated and validated against the live parser before it lands in your KB.
- "*Audit my KB for stale procedures.*" → dependency analysis with a list of unused elements and a recommended cleanup order.
- "*Render a dependency diagram for class tank-monitor.*" → inline Mermaid diagram in your IDE.

Where to Next

- **Developer reference & full setup:** g3mcp.ignitetech.ai
- **Product page:** ignitetech.ai/software-library/gensym/
- **Get access:** success@ignitetech.ai

© Ignite Enterprise Software Solutions, Inc. | All Rights Reserved