

Introduction

What derafu/content is and how its plugins fit together.

Anonymous · 21/08/2026

Where knowledge becomes product



Your content — documentation, FAQs, courses, blogs — is not just text. It's a valuable asset that can be packaged, reused, and leveraged like a digital product.

derafu/content is a PHP library that turns a directory of Markdown files into a content website: pages, navigation, tags, search and an interface AI agents can call directly. Every plugin adds one content type or one capability on top of the same core.

Two kinds of configuration

Every plugin has, potentially, two independent configuration surfaces, and it is important not to confuse them:

1. **Plugin configuration**, in the website's `services.yaml`, under `derafu.content.config.plugins.<name>`. This is set once by whoever runs the website (which directory to scan, which URL to call, whether a feature is enabled at all).
2. **Content frontmatter**, the YAML block at the top of each Markdown file (`---title: ...---`). This is set by whoever writes a specific piece of content (its title, tags, whether it's a draft, etc.).

Both are documented per plugin, but the frontmatter fields are mostly the same across every content type — see [Content frontmatter](#) for the fields shared by all of them. Each plugin's page only documents what it adds or overrides on top of that shared set.

Content hierarchy

Content types that support nesting (Docs, Academy, FAQ, Pages) build their hierarchy from the filesystem: a subdirectory represents a level of nesting, and it needs a file with the *same name* as the directory to represent that level itself. For example:

```
docs/  
  facturacion.md          # Represents the "facturacion" section itself.
```

```
facturacion/  
  anular-dte.md          # A child of that section.
```

Without `facturacion.md`, the loader has no content item to attach `facturacion/anular-dte.md` to, and the whole subdirectory is silently skipped — it will not show up in listings, search, or the API export. This is the single most common cause of “my content doesn’t show up.”

Missing content is a 404, not a 500

Requesting a URI that doesn’t exist (or a draft outside a local environment) throws `Derafu\Content\Exception\ContentNotFoundException`, which is mapped to **404 Not Found** — across every content plugin (Academy, Blog, Docs, FAQ, Pages) and [Storage](#) attachments alike, since they all resolve through the same `ContentRegistryInterface::get()`. A draft that isn’t allowed is treated the same way on purpose, so a 403 doesn’t reveal that it exists.

Plugins

- **Academy**: Course management (course → module → lesson hierarchy).
- **API**: Bulk JSON export of the content, meant for external indexing/RAG pipelines.
- **Blog**: Blog management.
- **Docs**: Documentation management.
- **FAQ**: FAQ management.
- **MCP**: Exposes the content as an [MCP](#) (Model Context Protocol) server, so AI agents (Claude Code, Claude Desktop, Cursor, etc.) can search and fetch it directly as tools instead of relying on stale training data.
- **Pages**: Standalone pages management, rendered under a `/pages` prefix by default so it doesn’t conflict with hand-authored flat pages.
- **Search**: Semantic search engine integration, with an optional LLM-based conversational answer (“ask”) grounded on the indexed content.
- **Sitemap**: XML sitemap of every indexable content item, for search engines.
- **Storage**: Attachment storage and download management.

Every content item can also be rendered as HTML, Markdown or PDF, on top of the JSON used by the API and MCP plugins — see [PDF and Markdown export](#) for what those two add beyond the raw content (video/quiz rendering, bundling a whole section into one file, download buttons on the HTML view).

Last updated on 21/08/2026