Toggly Docusaurus

Doc gating · Navbar links · Edge enforcement · No redeploys

Cheat Sheet
`@ops-ai/toggly-docusaurus-plugin`
Docusaurus 3+

Install

```
npm i @ops-ai/toggly-docusaurus-plugin
```

Plugin config

DOCUSAURUS.CONFIG.TS

```
plugins: [
  ['@ops-ai/toggly-docusaurus-plugin', {
    baseURI: 'https://definitions.toggly.io',
    appKey: process.env.TOGGLY_APP_KEY ?? 'YOUR_APP_KE
    environment: process.env.TOGGLY_ENVIRONMENT ?? 'Pr
    flagDefaults: {},
    isDebug: process.env.NODE_ENV === 'development',
  }],
]
```

App Key setup

1. Enable **Client Side API** in Toggly app settings.
2. Generate App Key — Type = **Frontend**.
3. Set `TOGGLY_APP_KEY` and `TOGGLY_ENVIRONMENT` in env.

Inline content gating

IN ANY .MDX FILE

```
import { Feature } from
  '@ops-ai/toggly-docusaurus-plugin/client';

<Feature flag="NewDocs">
  Visible when the flag is on.
</Feature>

<Feature flag="NewDocs" negate>
  Visible when the flag is off.
</Feature>
```

Page & navbar gating

FRONTMATTER

```
---
title: Beta Feature
x-feature: beta-feature
---

# Beta docs page
<!— entire page gated by flag —>
```

Plugin scans MD/MDX files, builds a route → flag map, and injects `__TOGGLY_PAGE_FEATURES__`. Client-side gate hides navbar links whose flag is off.

Client-side behavior

- Flags fetched server- and client-side.
- Navbar links hidden when flag is `false`.
- If flags can't be fetched, gate **fails open** — links stay visible.
- Client-side gating hides UI but content remains in HTML source.

Entity checks: core client

```
import { createTogglyClient } from
  '@ops-ai/toggly-client-core';

const client = createTogglyClient({ appKey });
const enabled = await client.getFlag(
  'PremiumBlock', false, {
    kind: 'DocSection', key: section.id,
    attributes: { Audience: section.audience },
  }
);
client.stopWebSocket();
```

Render the returned boolean in your own block. Plugin Feature has no entity props. Run browser checks in an effect and ignore stale responses; see the guide for the full lifecycle example.

Toggly Docusaurus

Doc gating · Navbar links · Edge enforcement · No redeploys

Edge enforcement

For true 404 / HTML stripping, pair the plugin with the Edge SDK.

OPTION A — CLOUDFLARE PAGES

```
# Copy functions/_middleware.ts from:
# github.com/ops-ai/Toggly.FeatureManagement/
#   toggly-docusaurus-edge-sdk/cloudflare/pages-funct

# Env vars (Cloudflare dashboard):
# TOGGLY_API_BASE_URL=https://definitions.toggly.io
# TOGGLY_ENVIRONMENT=Production
# TOGGLY_APP_KEY=<frontend-key>  (Secret)
# TOGGLY_PAGE_GATE_BEHAVIOR=404  (or redirect)
```

OPTION B — STANDALONE WORKER

```
# For GitHub Pages, Netlify, Vercel, S3, etc.
cd toggly-docusaurus-edge-sdk/cloudflare/worker
# edit wrangler.toml: route + ORIGIN_BASE_URL
wrangler secret put TOGGLY_APP_KEY --env production
wrangler deploy --env production
```

Which edge option?

- **Cloudflare Pages** → Pages Function (Option A)
- **Other hosts** → Standalone Worker (Option B)
- **Zero new infra** → Pages Function
- **Separate deploy** → Standalone Worker

Pitfalls

- Keep `flagDefaults` minimal — safe fallbacks only.
- Client-side gate fails open; edge worker fails closed (404).
- Restart `npm start` after config changes.
- Edge worker preserves last-known-good cache on transient fetch failures.
- Page-level `x-feature` is boolean — entity gates are not evaluated on the worker.

Resources

- Docs: `docs.toggly.io/sdks/docusaurus`
- Dashboard: `app.toggly.io`