ANTIDEFAULT

An embarrassing WordPress site, rebuilt from scratch and carrying the show years later

Case study — Doof! Media

Florian Ionescu

August 17, 2026

Client	Doof! Media (podcast)
Engaged via	As a direct client via my web agency
Industry	Media / podcasting
Stack	WordPress → SvelteKit, self-hosted on my own Talos Linux Kubernetes cluster
My role	Sole developer, from design through hosting
Engagement	Built 2024, in production and maintained since

Doof! Media came to me with an aging WordPress site: slow, generic, and a chore to update. They wanted a full rebuild, modern and fast, in one specific visual direction. Art Deco.

1 Deciding what to keep

The audit didn't take long, because almost nothing deserved to live. Not the design, not the admin panel, and definitely not the plugin graveyard holding the whole thing together. What I did have to respect was the content. Years of published episodes, sitting in a WordPress database that the plugins had spent years tangling on my behalf. Everything else went in the bin happily. The database I had to extract intact, and WordPress does not give that up without a fight.

2 Putting the site on a billboard

A podcast's website is a billboard for the show, so I made the website a billboard. Not a metaphor, a literal one: a frame, a lamp bolted to the top, and the light it casts down the front, all modelled in Blender. The pages render onto the billboard's face. I'd been given free rein past the Art Deco direction, and this is what free rein gets you. The SvelteKit side was almost boring by comparison. After a few weeks of fiddling, the new site was built and waiting on the content trapped in the old one.

3 Moving the content off WordPress

Rumor has it Stephen King is writing a book about migrating a WordPress database, and I believe it. The content you actually want is never in one place: it's spread across tables, half of it serialized inside fields the plugins invented, all of it shaped for WordPress rather than for getting out. This wasn't my first descent into madness, so I knew where to dig. I pulled Doof's full episode archive out and remapped it onto the new site intact. Took a week. A rough week, but a manageable one. Shortly after, the site went live on my own Talos Linux Kubernetes cluster.

4 Keeping it running

Since 2024 it's done the least dramatic thing software can do: nothing. It just runs. I host it and maintain it, and in roughly two years of production it's been down for about four hours total, each time resolved quickly. The cursed migration produced a site that's been boringly reliable ever since.

> *"People started mentioning the site to me, which had genuinely never happened before. The old one only ever came up when it was down."*
>
> *— Matt Freeman, Doof! Media*

5 If this sounds like your codebase

If your site or codebase is slow, embarrassing, and something you still can't do without, that's my favourite kind of problem. Message me and tell me what you're dealing with.