

A simple blog in markdown



CSDN @卷不动的小白

Below is a post about a blog I wanted to create in markdown, but as is the case with me, I changed my mind and went back to the old method. Access to this misfired idea is at this link.

https://ostrowski.net.pl/blog/index_copy.php

How is this blog created?

For a long time I have been testing various platforms with which I could put my thoughts on paper, but none of them was satisfactory enough for me. For one thing, I always wanted it to be hosted on my hardware, so platforms such as blogger were out. At the very beginning, I used blogger, which a long time ago (circa 2008-2010) was a very cool platform that allowed for relatively quick and easy editing of entries. Unfortunately, what happened to it was the same as with the whole Internet, instead of focusing on making it simple, easy and fun, blogger started to add various gadgets (blog monetisation, ads, telemetry, domain support, etc.), for people who are just starting their adventure with creating websites, it is relatively cool because it reduces the barrier to entry, but for people who know what they want, it only gets in the way. This is why it took me a relatively long time to find my favourite setup. The first stage

First, somewhere at the beginning of 2024, I realised that:

- GitHub is not for me
 - because I'm not a programmer, I may program a bit, but it's mostly for hobby purposes
 - I don't like my work being on some mega-corporation's servers.
- I hate applications that I don't understand how they work
 - I don't like applications that I don't understand how they work, which means we don't need all sorts of node.js packages, for me it's no different than running a simple .exe on my computer. Such a blackbox that starts and works, and what is inside is not your business.
- I don't want to host an application that's successful for a million users, scales well and so on.
 - I'm not a big company and I don't want every code or application I use to be ultra-scalable, I need something simple that solves a problem, not something that supports several different types of databases or has redundancy
 - at this point we don't really need MediaWiki, first of all it's a huge molecule but also the style is such that no matter who looks at it they will immediately see wikipedia
- editing of articles or content must be proportionally simple to the complexity of the content

- If I want to write a simple blog post with a few illustrations and maybe one or two lists, I don't want to create unknown sub-pages or go through 10 different forms.
- the language for defining the content must on the one hand allow for additional functionality (without exaggeration), but be simple enough to be able to write something in it straight away.

This is how I installed the DokuWiki package on my server, which to this day serves as my personal wiki, and every time I want to write something that is in a form, documentation for a project, or is a tutorial on how to build something, I upload it there. I really like the many functionalities of this platform and the fact that you can very easily (if you know PHP) add your own functionalities in the code.

If I feel like writing something that doesn't fit into the blog convention, I put it there. All the programmes I've written in my free time, all the tutorials, or maybe even the material from the times when I was still tutoring, I put there. What fits in better with the blog convention, e.g. my opinion on a given topic or a post about what I've recently done ... belong here.

But this blog is not on a wiki, it is here. My mind works in a slightly different way than I thought at the beginning. I thought dokuwiki would be a great place, there is a plugin for blogging so I have everything in one ... but not quite. I moved the blogs from my old blog to blogger, wrote a few new ones and somehow my enthusiasm faded.

I can't quite explain what caused this. It seems to me that because of the momentum of setting up my own wiki, when I wanted to put something in there it had to be „something“, I didn't feel comfortable with putting an entry in there describing something that might be perceived by someone as subjective, there were supposed to be purely technical texts that were documentation of something or explanation of something.

So in this way, there were more and more articles on the wiki, but the blog tab was empty. After that, I got into the habit of logging on to the wiki panel either to make a note of something so as not to forget, or to upload something interesting. I no longer had the creative energy to write something less technical in the form of a blog. It was a kind of bag into which I threw everything creative, but still somehow connected with electronics, IT or programming.

And here we come to ...

Stage two

The division into two pages was there from the very beginning of this pre-launch. There was the main page ostrowski.net.pl and the page for DokuWiki wiki.ostrowski.net.pl. The main page is a kind of landing page for those who find me on the web or click on my link, a description of who I am, what I do, and so on. Those who were more interested in what I do could go to the wiki page and see specifically what I do or what interests me. The home page, on the other hand, was empty, and for a while it was just an adverts page for tutoring, but after a while I thought I needed to give it some life.

I didn't want to lump the blog together with the wiki, so I decided I wanted to have a blog on the front page. My thought process was, it's always better to write a technical article and documentation of a solution on the wiki and then you can do a blog post discussing why you did something like that and what for. So that's what I did.

Of course, the requirements for such a system would not be easy to meet for modern blogging systems either, most of the requirements I have already mentioned in this blog post. In addition, I would also like to avoid something called the Static Site Generator, because I like the idea of creating a site this way and it won't be compatible with all hosts, but also compiling all the files every time you want to make a change to the site is absurd to me.

I could nest some off-the-shelf system into my site, but that's pointless because I would have all the disadvantages of that system nested in my site. Anyway, this is the opposite of simplicity.

After thinking about it for a few days, I decided that I wanted to write something ultra-simple of my own that would solve the problem. Of course, the language I chose was PHP. The structure of the program looks like this:

```
├── /blog
│   ├── posts
│   │   ├── post1.md
│   │   └── post2.md
│   ├── index.php
│   └── blog.php
```

Technical details will be published here [PHP: A simple blog using markdown](#)

As you can see from the file structure itself, it is intended to be simple concise and easy. The posts directory stores posts in the form of markdown files and the images are also stored there. The index.php file is used to scan the catalogue and create a list from which you can choose which posts you want to read, and blog.php is used to render them as a page.

It's a painfully simple solution, but it's what I wanted. It allows me to synchronise one directory on my laptop with the server and I can edit files in it as if they were files on my computer, but they are a representation of blogs on the site.

This minimises the threshold for starting to write, how much it costs to create a new text file in a folder, and how much it costs to prepare a whole technical description of the solution, to think about diagrams illustrations etc.

If I wasn't as lazy as I am ☹️ then a blog system for dokuwiki would probably be sufficient. But because I'm not a writer and it's a bit difficult for me to find the time and inclination to write something (and I'd like it to work differently), I have to look for solutions that make it as easy as possible for me to write something, and that's probably why I'm so dissatisfied and look for a better solution, one that at least is written by me and fits my website perfectly, because it was made for it.