Jekyll: make fast, secure static sites and blogs with Jekyll

udemy.com /static-website-generator-fast-secure-sites-blogs-with-jekyll/learn/v4/t/lecture/4572010

Why?

Websites need to be fast and agile

Websites today are becoming leaner, faster and more agile. And visitors demand quick and seamless experiences.

We pay a huge price for the underlying complexity of dynamic code running on a server for every request. This is a price we could avoid paying entirely when this kind of complexity is not needed.

We can always speed up websites with caching. No high-profile WordPress website would be capable of running without a plugin such as WP Super Cache. Large websites often rely on proxy caches such as Varnish, Nginx and Apache Traffic Server in front of their websites. The only thing is that caching is extremely difficult to get right. So, scaling a dynamic website can be very expensive!

And the truth is that even the most optimized dynamic website will normally be many times slower than a static solution. And if you have a popular site or blog nothing you do is ever enough.

Dynamic websites have made web a magnificent place but they have their pitfalls

Statistics say 70% of today's WordPress installations are vulnerable to known exploits (and WordPress powers more than 25% of the web). WordPress is used by 58.8% of all the websites whose content management system we know. This is 25.4% of all websites: http://w3techs.com/technologies/details/cm-wordpress/all/all

Some common problems Wordpress users face:

- My blog loading slowly.
- My blog crashing during peak periods.
- My blog becoming infected with malware.

WordPress is basically a content management framework. It can do just about everything and, if that's what you're looking for, then stick with it. I don't have anything against WordPress. It's great solution. I used it for many clients and still do. It's just evolved beyond its roots and beyond what I (and many other people) need. Jekyll, by comparison, is just a blogging platform. To some, this focus will be limiting, but if you only want to run a blog, then Jekyll has everything you need and nothing that you don't.

You won't be bogged down in options and insignificant choices, but you also won't lack the features you need most.

The writing experience is great. Your content is built with Markdown files so you can use the text editor you prefer.

What are static site generators?

The basic concept of a static site generator is: take dynamic content and data and generate static HTML/ JavaScript/CSS files that can be deployed to the server. Simple as that!

What is a static site (and what isn't)?

Static sites are only HTML, CSS, and JavaScript files. Of course, they also include images like JPEG and GIF,

graphic files like SVG and WebGL, or data formats like JSON or XML.

Static site les are **delivered to the end user exactly as they are on the server**. There is no server-side generation at runtime.

This means, for instance, that every visitor to your static site will be served an identical copy of index.html from the server until it is manually overwritten.

There is no server-side language like Python, Php or Ruby.

However, when speaking of static site generators, some are written using these languages but are intended to be run locally.

There is no database.

Because there is no server-side language to speak to a database, there is no database. But this does not mean that there is no data! There can be data stored as files or via an external service. This means that if you need common features like user registration/login, this would need to be via an external service!

Benefits of Static Sites

Performance

Static files can be served fast There is no server-side processing and no database to connect to, meaning that there is nothing to slow down getting a static page from the server to your end user. This also means that there are no bottlenecks that might cause slow requests when you encounter a significant traffic surge. Additionally, web servers will automatically set caching headers (such as Last-Modified) for static files, which reduce bandwidth usage.

Security

Static websites are immune to the most common attacks, as they do not contain dynamic content. There is no server-side language issues to exploit and no database to hack. Basically, as long as the files on your host are secure, your static site is secure.

Content versioning

Since your entire site, from configuration to content, is filebased, it is very easy to keep all aspects of it within a version control system like Git.

Simplicity

Building and maintaining websites is incredibly simple.

Versioning - Storing content in flat, text files, makes them ideal to be used with version control systems, such as Git.

Utilizing a static site generator means you can build websites with modern frameworks, and workflows, with a high degree of productivity and efficiency

Use markdown

Previewable: Because static sites are developed locally, you can play around with the site to your heart’s content without affecting the live site.

Hosting

Since no server-side language is required, hosting requires no complicated setup or maintenance, making it cheap and easy. In fact, there are even free options, like GitHub pages.

Despite these benefits, static sites, even with the help of a static site generator, are **not the solution for every type of site**.

What types of sites are static site generators useful for?

Static site work best for pages with **low degree of user interactivity**, pages that do not update frequently.

While static site generators are not built as a one-size-fits-all solution that support all sorts of customizability, there are common types of sites that meet the criteria above and therefore work well as static sites. The most common are:

- Blogs (most frequent use case for a static site engine)
- Informational sites
- Landing pages
- Selling pages
- Company pages
- Documentation (if you host documentation on Github you can benefit by having the community contribution, you can allow nput or pull requests from the user community)
- Portfolio
- And more

Use static website generators when you're **after simplicity** and want to put the **emphasis on the content**, which is what you should be doing. It doesn't really matter, whether you're focusing on hosting a huge website, or a simple portfolio, static websites can be extremely useful when it comes to saving money, time and resources.

But what about the forms, or comments or other dynamic part?

Even the most basic content site, like a personal blog, generally has dynamic aspects: commenting, feedback or contact forms and search, to name just a few

There are numerous services, both free and paid, that offer the ability to add dynamic aspects into static pages. These services were not specifically designed for static sites.

Markdown

Before we dig into static site generators, let me say a word on Markdown. Markdown has become a de facto part of the static site stack. Markdown is a simple way to add formatting like headers, bold, bulleted lists, and so on to plain text. It was originally designed to be an easy alternative to HTML, and allows people to create web pages with no HTML experience. It is a shorthand way to write HTML and is the default tool to write post and page content in most static site generators. It might be unfamiliar if you are not a web developer. But it can learned extremely fast (almost in a matter of minutes). Markdown's appeal is the simplicity of its syntax. Its philosophy emphasizes being easy to read first and and easy to write second.