

# Project: Migrate WordPress Blog to Jekyll 4



## Unix Tutorial RU based on Jekyll 4

I've finally decided to try running [Unix Tutorial RU](#), copy of this blog in Russian, in [Jekyll 4 CMS](#) instead of WordPress.

## Motivation for this Project

There's a few reasons that make migration to Jekyll a very interesting project for me.

Before you ask: **there's nothing wrong with WordPress and I'm**

**still a huge fan!** But I'm embracing more and more of git-based workflows for creating and deploying things lately, so it's a good opportunity to try blogging with git as well.

## **Improvements I Expect With Move to Jekyll**

Here's what I expect this project improves for me:

### **Much easier blog development and updating**

All the development is done locally on my laptop, with any changes done and debugged locally before committing them into local git or pushing to GitHub.

**Bonus:** there's no CPU/memory overhead for this development – so this is better and quicker than any VM, Docker (for WPengine dev) or even native httpd/mysql based WordPress setup.

### **Easier content editing**

Specifically, creation of new posts and pages in Markdown using my editor of choice – Sublime Text 3. It's possible to find plugins for Markdown editing in WordPress, but they are fairly limited, especially with the latest move towards block-based content editing.

**Bonus:** I can edit/write new posts without any connection to Internet (think WordPress desktop app either supports or plans to support it soon).

## **Better Code Examples and Shell Commands Snippets**

I don't know why, but process of posting syntax-highlighted snippets of code is more involved than necessary. Especially if one expects a preview right there in the WYSIWYG editor.

In Jekyll posting code is much simpler, even within Markdown text.

**Bonus:** because I'm editing the actual part of code that will be published, there's no plugin or WYSIWYG or block editor dynamic formatting risking that some syntax will be corrupted.

## **Easier images management**

I'm making a lot of screenshots and they use specific naming convention. If I upload them as is – files named like 2020-03-08-22-44-00-screenshot.png – it becomes rather tricky to later rename them for better SEO performance. With Jekyll renaming an image is as simple as renaming the image and updating Markdown page linking to it.

**Bonus:** any image uploads are as quick as moving files on my SSD drive! But yes, I obviously need to upload them into

GitHub repo later.

## Historical tracking of changes

This is becoming a very important point for me – it helps so much to keep track of all page updates as git commits! It's possible to see latest update and previous versions of a page in WordPress as well, but there's no easy way of seeing a timeline of all changes with description of changes. I'm using Jetpack status updates right now in WordPress environments and it helps, but you get only a week of history and even then you get a list of pages updated, but not the list of commit-style descriptions of what you changed.

Since I'm hosting Jekyll blogs on GitHub, I get to enjoy history, roll-back and automated deployments. Awesome!

## Project Plan for Migrating WordPress Blog to Jekyll

These are the elements of migration as I see them:

- setup new [Jekyll blog and host it on GitHub](#)
- create sidebars
- implement linking to original posts/pages on [Unix Tutorial](#)
- recreate the same permalink structure for posts and pages
- migrate content for the first few posts and pages

Most of the high-level steps are generic enough, but here are notes of the modifications I made to date.

## Implement Linking to Original Pages in English

I added a few new variables to the [Front Matter](#) section of page/post code:

```
translation: https://www.unixtutorial.org/sublime-text-3
translation_text: "Sublime Text 3"
translation_flag: /images/en-flag.png
```

In `_layouts/post.html` I added the following section to make sure of these variables:

```
{% if page.translation %}
<strong>Перевод</strong>: <a class="c-post-card-recent__flaglink" href="{{ page.translation }}" target="_blank"> {{ page.translation_text }}</a>
{% endif %}
```

## Current Progress on this Project

Here is what I got done over this weekend:

- setup new Jekyll blog
- created private GitHub repo (can't make it public yet because I'm using a paid Jekyll theme)

- linked GitHub repo to my free Netlify account
- created sidebar widgets
- updated Jekyll theme to match top menu of [UnixTutorial.org](https://UnixTutorial.org)
- manually created the first few pages
- created first few posts
- implemented support for translation links (back to [Unix Tutorial](#))

## Possible Next Steps

- Make GitHub repo for [Unix Tutorial RU](#) a public one
- Further develop sidebar widgets
- Integrate with Pateron or similar service

## See Also

- [Unix Tutorial Projects](#)
- [Project: Setup Jekyll Blog and Host it on GitHub Pages](#)