Projects on my Homepage

To this date my website always featured a ***Projects*** tab. Now I decided to remove it. The reason is, that it is just a little annoying to keep up to date.

I don't want to update it every time I drop a project and also some old projects don't really look good between the newer ones. It's not like I want to remove my history, but when someone finds my homepage they should see what I do *now*.

Instead I will just write a blog post on everything that I find noteworthy. And with this post I should give a short heads up on what would go missing otherwise.

So here are all the projects that you can't find via the projects page now.

dotfiles

Since May 2019 I have been managing my desktop/laptop/server configuration via a dotfiles repository. It has gone through quite a few changes and I guess this is a good time to show off some of the highlights.

- Different Linux distributions are managed with branches. The `arch` and `nixos` branches are forks of the `master` branch.
- I previously used i3, but now I have an extremely customized xmonad configuration. I also spent way too much time on xmobar.
- Vi-like tmux configuration.
- Mutt, offlineimap and msmtp for my email.
- Custom terminal color scheme.
- I was using vim, but switched to neovim lately.

I have thought about switching to home-manager, since I mainly use NixOS nowadays, but didn't pull through yet.

haskeme

This is a small program, that translates I-Expressions into S-Expressions for lisp-like languages. It's also a perfect example of a project I'm not particularly proud of now. It's advertised as a compiler, when it's really just a quick hack. If I were to do it again, then I would just write an actual tokenizer, parser and AST.

blugon

Blugon is probably my most popular project to this date. It's a blue light filter written in Python following the unix philosophy of doing just one thing. It's also pretty easy to set up with a man-page and a nice readme.

go

One of the hardest project I tackled is an HTTP webserver with a javascript frontend using websockets to communicate with the server. I wrote the server with servant and the frontend using miso, which makes it a *100%* Haskell project aside from CSS and nix deployment.

It's a multiplayer game and is pretty much fully functional, it just lacks some minor improvements which I'm too lazy to do now. Ah, and it's sort of missing a scoring system…

So if you want to play go feel free to try it out. You'll just need a machine runing NixOS.

blucontrol

I already wrote blugon, but Python didn't feel like the right language. After diving deeper into Haskell, I noticed that it's quite a good fit. This application is abusing monads and makes pretty much every part configurable. It is configured similar to xmonad, but uses many newer features of GHC. I actually want to write a showcase on it, but didn't get to it yet.