Arūnas Butkus EKSfmu-16

Laboratory work 7

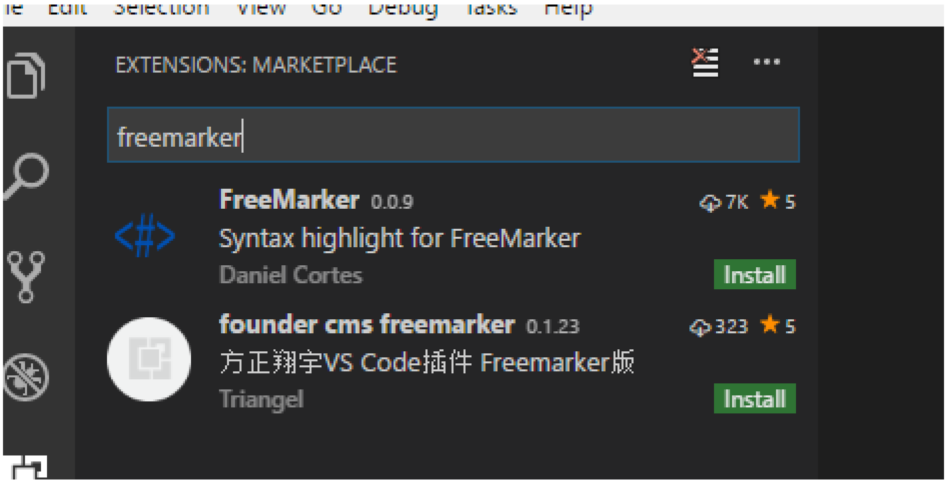
What I did:

Part 1: Open source text/document editors

For paper writing and such there are few options and specifically thinking about MS Word alternatives the first, and mostly only, thing that comes to mind is OpenOffice/LibreOffice suite with its text, spreadsheet and etc. editors. Those are under GNU license as they spawned from need of document editors on Linux based platforms. There isn't really any significant alternatives in terms of document editing and it is not likely that any will appear since now there are multi-platform free online editors from Google and Microsoft themselves.

Though when it comes to coding, it's still open season. As MS Visual Studio, Eclipse, etc. becoming large programs with a bunch of unnecessary features in some applications more lightweight, quicker editors are gaining popularity. Some coding can be done with Windows' Notepad – wrote quite a few .bat files with it myself, but is really not an editor for a project of any significance. Notepad++ showed that with some predictive coding, highlighting and in certain cases syntax checking, simple lightweight program can be a nice and fast way to code. Wrote a few C/C++ programs with it since I didn't want monsters like MS Visual Studio, or iffy free C IDEs like CodeBlocks that didn't properly work on Windows 10 most of the time (used and liked it on Windows 7). By the way, CodeBlocks is a free IDE with GNU GPL/LGPL license.

So this is where my current favorite midsized project editor for front-end development comes in. Visual Studio Code – Microsoft endorsed MIT licensed editor with most of the goodies available from old big programs (IntelliSense, highlighting, debugging, git integration). In latest project had to work with freemarker files – eclipse had no highlighting, no error checking; tried opening the project with VSCode, just needed to quickly search and add freemarker support [1] and now it is much more convenient than eclipse. It also has terminal/command line integration and is actively developed right now. Personally, found it really convenient in Angular development.



freemarker

FreeMarker 0.0.9 7K ★5
Syntax highlight for FreeMarker
Daniel Cortes

founder cms freemarker 0.1.23 323 ★5
方正翔宇VS Code插件 Freemarker版
Triangel

Fig. 1. Adding extensions to Visual Studio Code

Soon after starting working with VSCode I saw that some use Atom – really similar alternative to VSCode. It is also MIT licensed and freely available on GitHub. Didn't test it out, but looks near identical in execution. Both of these predominately deal with typescript though as I've seen VSCode has capabilities in other languages as well and Atom supposedly is similar in that regard. According to the site I'm getting all these open source programs, Atom is mostly a JavaScript editor.

In case of JavaScript Adobe has an open source program to offer – Brackets. It is under MIT license and offers lightweight customizable HTML, CSS, JavaScript editor with emphasis on seeing changes quickly in its browser view. There are a few other for JS, especially if you are inclined to work in terminal for some reason. Things like Slap – sublime-like terminal-based text editor (MIT license). Then there's Caret – main feature being that it works as a Chrome extension (GNU GPL license). Zed – mostly advertising multiple cursors and its bare bone-ness (MIT license). Also can work standalone or as Chrome app…

And this is laboratory work, not a course project, so enough waxing lyrical. Here's a short list of the rest:

- Lime – Sublime Text alternative under 2-clause BSD license
- textmate – C++ editor with GNU GPL license.
- neovim – "improvement" of vim editor for Linux (Apache 2.0 License).
- ShareLaTeX – LaTeX editor – GNU AGPL license.
- MiKTeX – what I use for LaTeX writing (no license: "To the best of my knowledge, all components of the MiKTeX software are freely redistributable" - Christian Schenk)
- Komodo Edit – python editor (MPL 1.1 license).
- Leo-Edit – python editor under MIT license.
- Spacemacs – emacs editor (whatever that is) under GPLv3.
- SpaceVim – another vim editor that is an extension to vim and neovim (MIT license).
- LightTable – MIT licensed Chromium browser alternative.

Part 2: Open source spreadsheet editors

Other than the ones that come with LibreOffice and OpenOffice I don't see any other noteworthy candidates that are open source. LibreOffice has a mess of licenses mostly under MPLv2.0, but including code licensed under Apache 2.0, GPL and LGPLv3+. OpenOffice, nowadays "Apache OpenOffice", as the new name suggests uses Apache 2.0 license.

Part 3: Open source presentation editors

Similar situation to a degree, leading programs come with LibreOffice and/or OpenOffice. Though, apparently, Calligra suite is good too. GPL/LGPL licensed program, originally part of KOffice, contains all the crap that other Office type suites offer including presentation editor, so might as well try that I guess.

Part 4: Open source image editors

Most obvious choice is main Linux image editor GIMP. Now works on all platforms, even Solaris (whatever that is). Licensed under GNU GPL v3+. Attempted Adobe Photoshop alternative for early Linux days when Photoshop was PC only. Feature list nowadays is quite bare bone compared to good editions of Photoshop, but does have a decent plugin support with a bunch of plugins to choose from.

Next I stumbled across digiKam – seems more like an image manager since main touted features are image libraries, configurable automatic import/export, conversion to/from RAW and any other image format and easy metadata editing with some editing of image itself tacked on at the end. Licensed under GNU GPL, fully fleshed out and still in active development for all main platforms with latest release on September 12th.

Then there's ImageMagick – image editor without a GUI: "The functionality of ImageMagick is typically utilized from the command-line". Though, obviously, it does have some GUI extensions since casual user is not going to like command-line image editing. On their page they list 20 editors written in different languages form known ones like C or Java, to old ones like Pascal, to ones I hear for the first time like Tcl/TK or Lisp. Main features touted are that it can create images dynamically and "automagically" and that its multithreaded computation has increased performance when working with mega-, giga- or tera-pixel image sizes. Uses Apache 2.0 license, by the way.

"Krita is a free and open-source raster graphics editor designed primarily for digital painting and animation purposes". While initially it was just an editor bundled with other office software and then evolving into pretty much its own program what competed directly with Photoshop and GIMP, now it is marketed more to digital painters going into same market as Corel Painter or SAI. GPLv2+ license.

Speaking of, Paint Tool SAI used to be free at some point. When I first tried out digital painting I used SAI and it was free, now it is shareware – 30-day free trial and then you're encouraged to buy it so it is not open source no more.

Instead of raster images there's the vector graphics editor alternative – Inkscape. Always though it is MACOS trash, but apparently now it supports all main platforms. Uses GNU GPLv3+ license. So as mentioned its main shtick is that it user vector graphics instead or raster – manipulates shapes that are made of vectors, not drawn pixel by pixel – somewhat like pen tool on Photoshop, but the whole program is that just more fleshed out.

And to finish off, MS Paint bigger cousin – Paint.NET. Till version 3.36.7 was open source GPL licensed and now it is "freeware" since it moved to Windows Store… It looks like MS Paint, has its features but with a bunch of windows with extra features tacked on – layers, color picker, multiple projects, floating toolbar.

Conclusions

Honestly, few pieces of software I had to install for embedded systems laboratory works seriously annoyed me so I'm not adding more trash to my PC anytime soon. So this report is written with MS Word – free edition thanks to KMS – and I will continue using it for document related work. Though, I am using MikTex for my master thesis – latex editor. For spreadsheets – MS Excel – don't need it often and it's good enough. For presentations – MS PowerPoint – same reasons. If I ever going to need alternatives for any of these I'll just use online ones since everything seems to be going online these days and from what I tries online version of MS Word or even Google Docs are serviceable already.

When I comes to image editors, now I barely use any. MS Paint services most needs. I take screenshots with ShareX so most of the times I don't even need to crop the image. Though if something more fancy is needed Adobe Photoshop is present. Though I might explore what Krita has to offer, mostly because I liked its cover image. Or ImageMagick – to see what the deal is with image editing using command-line. And if I'll ever return to digital drawing then I'll got for something like Inkscape – drawing with vector graphics is nice since it can autocorrect your shaky hand (though issues appear when working with small details, hashing and other many vector creating tidbits).