Ronald Casili

nvlled@gmail.com
+639052182048
https://github.com/nvlled

About

I am hobbyist who has a wide range of interest, including math and programming. I am not openly competitive, but for purposes of knowing my current edge and limitations I do try to measure myself against other people who are in the same competing space. I still have lots to learn, but I also actively don't try beat myself too much up if I hardly make an inch towards my mutable goals.

Education

Bachelor, Information Technology - University of Pangasinan Urdaneta 2018

Skills

- **Basic System Administration/Maintenance**
  - Debian/Linux
  - Tools/Utility scripts with bash, python and node
  - Remote administration with SSH
  - Version Control with Git
- **Desktop Development**
  - Python3 with PyQt5 (cross-platform)
  - Node with Electron (cross-platform)
  - C#/VB.net with WinForms
  - Java with Swing
- **Web Development**
  - Node/Express
  - PHP7 (plain or with laravel)
  - Mono Asp.NET
  - Go backend
  - HTML5/ES5/CSS3
  - jQuery or plain DOM API
- **Others**
  - Basic 2D game developement
  - Library Development
  - Writing (Technical or Creative)

Recent Projects

Doctrac

A university document tracking system. It remains to be seen what the future of this project is since partly the stakeholders couldn't form a definite consensus on the process, and partly because I couldn't deliver on designated times.

Technologies

- **PHP/laravel** – This is only the second time I've used laravel, the first one was for a project assignment, so my code wasn't idiomatic to the conventions used by the laravel developers. The major pain point I experienced with laravel is the java-ish design pattern coding conventions used. I personally find it needlessly complicated, and more often than not it was tedious to adapt the framework to my problem.

- **jQuery** – I'm actually more of a vanilla js guy who uses DOM API directly, but I've used jQuery because the would-be maintainer is familiar with it. For quick page interactivity, jQuery is a good tool, but for really complex UI, it starts to devolve to spaghetti code.

- **domvmjs** – I rewrote a really complex and messy jQuery UI code in domvm, which made the code simpler and maintainable. There are lots of trending vdom libraries now (such as react and vue) but I chose this one since it's a small library with no dependencies.

- **Bootstrap 4** – I actually rewrote the site interface design twice before finally succumbing to bootstrap. Interface and layout design is my weakness, and it's hard to it right even with css grid and flexbox.

Cbtest

A (web) computer-based test and exam for technical drafting. Notable features include automatically grading drafts by comparing images, and automated test score grading and compilation.

Technologies

- **node/express** – I've used this for the backend, which is a standard choice since express is a good enough for most backend stuffs. I'd like to get a chance to develop SPA with react or vue and see what the developer affordance and ergonomics they provide.

- **HTML5 with pug and CSS Grid/Flexbox** – I learned and tried using CSS grids since the client had specific requests for layout design. I've managed to fulfill the request, but working with CSS grid was cumbersome since it behaves differently on browsers. But it's possible that the layout was really complex or that I haven't fully learned how to utilize CSS grids properly.

- **Plain JS/ES5** – For the client scripts, I don't need much page interactivity so I resorted to just using plain DOM API and ES5 for optimal compatibility.

- **iSpring Suite 8** – This is a plugin for MS Powerpoint for creating interactive test/exam materials. This is what the client was already using, and I needed to implement automation for the manual processes. I needed to programmatically modify the code generated by ispring, but it does not provide an SDK, so I just did a little reverse engineering on the base64 blob embedded in the script.

- **Image Diff Library** – I forked and modified an image diffing library so I could count the number of pixel differences between two images.

Trending scanpath analyzer

This is a work from a PhD student who needed automation from the current manual MS Excel computations. The problem domain involves taking a group of files generated by an eye-tracker device, and then finding a trending scanpath from a group of student participants. The full problem domain is a bit out of my knowledge scope, but my implementation have a formal document/specification to refer to.

Technologies

- **Python with PyQt5** – I don't have much exposure to python aside from when I was learning to program, but it's a nice language/tool that I definitely use more in the future. I had a bit of trouble setting up the python environment, but I managed through with virtualenv, which works seamlessly with windows and linux.

- **Qt5 with Qt Designer** – I don't have much luxury time learning another layout system, so having WYSIWYG interface tool such as Qt Designer is a big help.

- **Pyinstaller** – I used this to make standalone executable binaries.

Dilim

Digital Library Management. This is a work from my sister who is a highschool teacher and is assigned on managing a library. Her primary complaint is creating DDC call numbers for the books.

Technologies

- **Electron JS** – Electron is a nice technology in creating cross-platform apps, but I sort of dislike the bloated application size.

- **NPM Libraries** – I had to write two libraries for the generation of DDC call numbers since there were none in the npm, the one I could find was a windows GUI app.