Paul Olteanu

647-449-6597 | p.a.olteanu@gmail.com | github.com/PaulOlteanu

Work Experience

Nymi Inc   *Toronto, ON*
SDK Developer   *September - December 2018*

- Developed 3-part async SDK supporting multiple Nymi Enabled Applications and Nymi bands, usable in any language supporting a **C FFI**
- Interfaced with C security library to encrypt communications between NEAs and Nymi bands
- Optimized **Phoenix** server communications to minimize latency in **Citrix** environments
- Consumed **Windows APIs** to support applications running locally with the Nymi Runtime on Citrix servers
- Key Technologies: **Rust, Elixir + Phoenix, Windows APIs**

#paid Inc   *Toronto, ON*
Full Stack Developer   *July - August 2017, January - April 2018*

- Architected parallelized job pipeline using Elixir's **GenStage** library for worker backpressure and rate limiting
- Processed **30,000+** users/day, allowing for better campaign targeting and the discovery of new influencers to onboard
- Configured **AWS EC2** instances for staging and production use, and set up **Squid** proxy servers for request caching
- Led project updating multi-step form for clients to create campaign proposals, increasing amount of data collected from RFPs
- Key Technologies: **Ruby on Rails, Elixir, AWS EC2 & IAM, Postgres, Mongo**

Projects

Software Renderer   *February 2019*

- Supports line and triangle rasterization
- Multiple projections including orthogonal and perspective
- Implemented in **Rust**

Ray Tracer   *December 2018*

- Multiple cameras able to be arbitrarily positioned for every scene
- Supports multi sample antialiasing (MSAA), and both Lambertian and specular reflecting surfaces
- Able to capture and export scenes as PNGs
- Implemented in **Rust**

CHIP-8 Emulator   *October 2017*

- Handles all 35 opcodes + legacy mode for opcodes that have traditionally been misimplemented
- Interprets binary .ROM files (CHIP-8 machine code)
- Implemented in **C** using **SDL2**

Version Control System   *January 2017*

- Supports commits with messages, reverting, and full logs
- Stores commits as diffs to reduce storage footprint, with complete backups every *n* commits to increase revert speed
- Implemented in **Python** using **Click**

Proplist   *October 2015*

- Contributed to library adding binary (string) key support to keyword lists by re-implementing the Keyword API
- **Published** on Hex (Elixir's package manager), with **~1,000 installs**
- Implemented in **Elixir**

Education

University of Waterloo   *Expected Completion Apr. 2022*
Candidate for B.AS. in Computer Engineering. Currently 2A