30 Livonia Place, Apt 510
Toronto ON M1E4W7

Nathan Jaremko

(705) 943-2631
njaremko@gmail.com

Employment

QA Analyst **PointClickCare** **August 2015 – April 2016**

Integrations

- Discovered security vulnerabilities that enabled SQL commands to be executed from text boxes in the application. Reported them to the team lead, and a fix was released in the next emergency patch.
- Designed and implemented unit tests for critical components of the application.
- Performed regression, backward compatibility, functional, and integration testing.
- Participated in daily scrum stand up meetings, as well as weekly backlog grooming and sprint planning sessions.

Game Development Intern **S.S.M. Innovation Centre** **June – August 2012**

- Contributed as a member of a team to meet and exceed project deadlines demonstrating strong organizational skills.
- Designed a competitive race game in the Unity3D game engine using C# and JavaScript to implement AI, movement, level generation, animation, and physics
- Extended engine systems in C++ to support A* pathfinding using the Manhattan distance heuristic and creating dynamic navigation maps using ray casting
- Wrote unit tests, and tested product regularly to ensure quality of development
- Strengthened communication skills by interacting with team members, project manager, and supervisors

Education

Toronto, ON **University of Toronto** **Fall 2014 – May 2018**

Candidate, Bachelor of Science

- Computer Science Co-op Specialist in Software Engineering, May 2018.

Technical Experience

Projects

- **Abot** (2016). Open-Source virtual assistant that is user extensible. It currently supports basic functionality like setting reminders, checking the weather, and searching for nearby restaurants. Abot currently supports communcation through email, text message, command line, Twitter, and Slack. It currently processes messages in less than 5ms, and runs on machines with 512MB of RAM. Golang, SQL, Bash
- **Knuth text algorithms** (2016). Open-Source golang library implementation of the Knuth-Plass line breaking algorithm, and the Knuth-Liang hyphenation algorithm. Golang
- **Python SQL-like querying** (2015). Created a database querying program with the ability to search and retrieve data, as well as produce Cartesian products of tables. Functionality mimicked SQL with the use of SELECT, FROM, and WHERE tokens. Python
- **Regex permutations** (2015). Created a program that returned a list of all permutations of a string that were valid regular expressions, and represented them as a tree. Recursively implemented a function that checked if an inputted string matched a given regular expression. Python
- **Skip List** (2015). Implemented a skip-list in Python using singly-linked lists, then used it to implement the multiset ADT. Python
- **C++ Algorithms** (2015). Created C++ implementations of linked-lists, stacks, trees, queues, and hash tables. As well as, the 3 partition quicksort and mergesort algorithms. C++

Languages and Technologies

- C; Java; Go; Swift; Haskell; Ruby; Python; Elixir; C#; C++; SQL; JavaScript;
- Vim; Linux; Visual Studio; Microsoft SQL Server; IntelliJ; XCode;