LUISA ROJAS

COMPUTER SCIENTIST

hello@luisarojas.com ✉
github.com/luisarojas
luisarojas.com 🌐

EDUCATION

MSc in Computer Science
September 2017 – *November 2019 (expected)*

University of Ontario Institute of Technology

BSc in Computer Science (Honours)
September 2013 – April 2017

University of Ontario Institute of Technology

EXPERIENCE

Teaching Assistant
January 2017 – Present

University of Ontario Institute of Technology, Faculty of Science

Provided academic support to junior and senior Computer Science students for the following courses at Ontario Tech University:
→ Programming Workshop (C++)
→ Software Quality Assurance (C++, Java)
→ Web Development (JavaScript, MongoDB)
→ Data Structures (Java)

System Administrator
May 2014 – December 2014

University of Ontario Institute of Technology, International Office

- Designed and maintained the International Office website.
- Oversaw the international students' University Health Plan (UHIP) program, which covers 25% of the student population.
- Aided in the planning and marketing processes for various events held for international and exchange students.

LANGUAGES

English (Fluent), Spanish (Native), French (Basic)

SKILLS

Programming
C, C++, Java, Python

Databases
PostgreSQL, NoSQL, MongoDB, Neo4J

Web Development
jQuery, NodeJS, Flask, HTML, CSS, Bootstrap

DevOps
Git, GitHub, Docker

Machine Learning
Keras, TensorFlow

Source Code Instrumentation
TXL

LEADERSHIP

Director of Graduate Studies
Ontario Tech Student Union, Board of Directors

Computer Science Program Representative
Ontario Tech Graduate Student Council

Vice-Chair and Senior Mentor
Ontario Tech ACM-W Student Chapter

Founding Chair
Ontario Tech ACM-W Student Chapter

Vice-President of Marketing and Design
Ontario Tech Science Council

Mentor and Judge
Local Hack Day by GitHub and MLH

Panelist and Logistics Volunteer
Go Code Girl 2019 by Ontario Network of Women in Engineering

PROJECTS

Automatic Concurrent Fault Localization
April 2018 – Present

This project is being developed as my master's research thesis. To address challenges associated with multithreaded programming, I have created an automatic fault localization tool for concurrent Java programs. The tool utilizes a combination of noising strategies and a heuristic search algorithm to identify code blocks with a high probability of being buggy.

Docker   Java   Python   TXL

Mentor-Mentee Matching System
February 2018 – April 2018

Designed and implemented a web-based platform for a matching system between mentors and mentees for the Peer Mentorship Program at Ontario Tech. Given a set of survey answers by potential mentors and first year students, our system matches them by faculty according to their compatibility. It outputs the results in an Excel sheet for convenience and stores them internally for visualization and future use.

CSS   Docker   HTML   JavaScript   Neo4J   PostgreSQL   Python

Distracted Driver Detection
December 2017

Trained a deep Convoluted Neural Network (CNN) to predict with 99.4% accuracy whether an individual is distracted-driving as well as what type of distraction is involved. The VGG16 CNN model, pre-trained on ImageNet, is coupled with a retrained fully connected model and tested on previously unseen distracted driver images.

Keras   Jupyter Notebooks   Python   TensorFlow

Threaded Paws
October 2016 – October 2017

As part of my undergraduate thesis project, I designed and developed a serious game that can assist students in learning different concurrency concepts and pitfalls, such as thread interleavings, data races, starvation and deadlocks. Threaded Paws was designed for second to fourth-year Computer Science and Software Engineering university students at the undergraduate level; it will be used in the classroom for the first time in September 2019.

C#   Unity

dynOBD
October 2016 – December 2016

Android mobile application that records and keeps track of statistics about any given car trip. It uses a Bluetooth connection to the vehicle in order to provide trip information and other live data visualizations to the user, such as current speed and vehicle throttle. The data is collected using different APIs, like Google Maps and OBDII Java, and is subsequently stored in a local SQL database for later access.

Android Studio   Java