Long Pham

Software Engineer - Systems Test Engineer (L2)

Burnaby, BC, CA
778-789-7599
longmpham93@gmail.com
longpham.ca
github.com/longmpham
linkedin.com/in/longmpham

EXPERIENCE

OSI Maritime Systems Ltd, Burnaby, BC, CA — *Systems Test Engineer (L2)*

NOV 2019 - PRESENT

- Read and understand customer requirements, consult in design implementation, and write test procedures as per customer requests
- Run tests and verify the requirements using multiple testing techniques like blackbox, regression, and rigorous manual testing
- Created small scripts to alleviate manual testing
- Naturally enrolled as a young leader and acted as a team lead for co-op students and junior engineers (L1)

Western University, London, ON, CA — *Teaching Assistant*

SEPT 2016 - APR 2019

- Lead instructor for multiple courses including (1) microcontrollers, (2) software design, and (3) operating systems.
- Provided office hours, guidance, and organization between labs
- Role model for over 500 cumulative students and fellow TAs.

Western University, London, ON, CA — *Software Instructor*

MAY 2016 - AUG 2016

Instructed and created a simple HTML, JavaScript and CSS summer course for 25 elementary students in the summer using the Phaser.io

EDUCATION

Western University, London, ON, CA — *Masters of Engineering Science - Software Engineering*

SEPT 2016 - APR 2019

After 3 years of studying, I graduated with a thesis published by the university in the research of deep learning using Gabor filters to increase the speed and efficiency of convolutional neural networks (CNN).

SKILLS AND ATTRIBUTES

Software Development Lifecycle

Coding (Python, C, JavaScript)

Leader

Fast Learner

Problem Solver

Dependent

AWARDS

NSERC Research Grant (2019)

Certificates

Canadian Clearance (5 Years)

NATO Clearance (10 Years)

LANGUAGES

English (Primary)

Vietnamese (Secondary)

Western University, London, ON, CA — *Bachelors of Engineering Science - Software Engineering*

SEPT 2012 - SEPT 2016

Built a foundation in engineering learning the software development lifecycle, while gaining knowledge with essential CS fundamentals.

PROJECTS

CNN Classifier — *Using Gabor Filters as the first layer to introduce speed and efficiency vs minimal accuracy loss*

Thesis work that focused on the implementation of the Gabor filter used within a 7-layer convolutional neural network to classify 4 datasets consisting of almost 200,000 images in total. This network trained faster than other networks, but resulted in a slightly lower accuracy in comparison to other networks.

Personal Website — *Website Created with MERN (ReactJS)*

Using the ReactJS library to build a frontend single page application that shows my personal achievements, past history, interests, and my resume.

Meme Generator — *ReactJS Frontend App*

Using React with hooks, fetch, async/await to grab images from the public API from ImgFlip

Profile Viewer — *MERN App for user auth and minimal frontend*

Created a MERN app for basic user authentication using JWT and Bcrypt for password encryption. This app also provides a basic frontend that shows a user login page along with a profile page that shows their stats.

Impatiens Nails Website — *Website Created with ReactJS*

Building a frontend Nail E-Commerce App from scratch. These nails are custom made and designed by my significant other and we plan to start a business from the ground up selling custom press-on nails

Click The Boxes — *ReactJS Frontend App*

Created a React app to click black boxes to reveal the image behind it. Used React hooks to implement this.

To-Do (CRUD) App — *Full stack app using MERN*

Created a basic CRUD MERN app for a to-do list. Barebones, first implementation of learning how CRUD works.

Google Image Binary Classifier — *Python Scraper + Keras Deep Learning Framework*

Learning how to create a binary classifier pulling images from Google Search, creating a dataset from that and using Keras as the training network to classify the images.

Basic Watch — *DE-10 Development Board Wrist Watch*

Created a simple watch using the DE-10 development board programmed to use an ARM processor. The watch was able to compute time in milliseconds, seconds, minutes, and hours. The clock featured basic stop, reset, and lap capabilities. This was used for a lab for over 300 3rd year engineering students to practice assembly language as well as C.

Phasio.io Shooter Mini-Game — *Phaser.io mini-game using the Phase Engine*

Created a simple shooter using PhaserJS. Created simple AWSD controls using the mouse pointer to aim and shoot with. Extras such as high scores, lives, and weapon boosts were implemented as features. This game was created with JavaScript and was later used as a curriculum for 25 Grade 8 students interested in coding.