

Edmond Chuc

<https://www.edmondchuc.com> | <https://github.com/edmondchuc>

ABOUT

I am a passionate software developer who has recently graduated. I love working on challenging and technical projects. I believe in knowledge sharing to help improve my own skills and those around me. I have a high level of initiative, often working on my own projects or doing readings in my spare time. My current interests include the web, web architecture, Python, test-driven development and continuous integration.

EDUCATION

Griffith University	Gold Coast	SAE Institute	Gold Coast
<i>Bachelor of Computer Science</i>	2018	<i>Bachelor of Interactive Media</i>	2018
Major in Software Development		Major in Games Design	

EXPERIENCE

CSIRO Land & Water	Brisbane
<i>Software Developer</i>	July 2018 – present

I am part of the technical infrastructure team in constructing Australia's first distributed, large-scaled Linked Data network system to solve data interoperability limitations across government organisations.

- I helped recreate the functionality of CSIRO's [Persistent Identifier Service](#) in Apache HTTP Server as a light-weight alternative.
- I implemented new persistent URI identifiers for the [Australian Government Linked Data Working Group \(AGLDWG\)](#).
- I configured a [Geoserver](#) instance to serve the [Bureau of Meteorology's Geofabric dataset](#) via Web Feature Service and Web Map Service APIs. The features of the Geofabric dataset were visualised through the WMS overlaid on top of a styled [OpenStreetMaps](#).
- I deployed web APIs that served spatial datasets as Linked Data.
- Comprehensive endpoint tests for each Linked Data dataset API service.
- Technologies: Python, Flask, Pytest, HTML, CSS, Linked Data, Semantic Web, Geoserver, Web Feature Service, Web Map Service, Apache, PostgreSQL, AWS.

CSIRO Land & Water	Brisbane
<i>Student Industry Placement</i>	February 2018 – May 2018

I updated a Linked Data web application called LODE. Its sole purpose was to extract OWL properties and display them as a human-readable web document. I fixed many of its templating issues and reconstructed the architecture of the application.

- Previously the architecture relied on Apache, PHP, jQuery and Java. It now only requires Java.
- I implemented WebVOWL as an optional ontology visualiser within the HTML document.
- It is now actively used by the AGLDWG and the Linked Data community at [LODE2](#).
- Technologies: Apache, PHP, jQuery, Java, Java EE, Maven, Tomcat, Jetty, Linked Data, Semantic Web, AWS, HTML, CSS, XSLT, PHP.

PROJECTS

Portfolio Website

<https://www.edmondchuc.com>

I created a portfolio website showcasing some of my work in greater detail. I also created this website to learn more about web technologies such as HTML, CSS frameworks and using JavaScript for some basic form validation. The back-end consists of a small Flask web application which handles requests sent from my contact form. These requests are then sent to the SendGrid service which then sends an email to my inbox. Apache HTTP Server is used to serve the static content. The back-end runs on an Ubuntu Droplet from Digital Ocean.

- Responsive mobile-first approach. Technologies: HTML, CSS, JavaScript, Python, Flask, Apache

Finding the Global Minimum Energy of N-Atom Bonded Molecular Models

Paper: <http://www.edmondchuc.com/model-molecular-conformation/paper.pdf>

A research project using L-BFGS and a genetic algorithm to find the global minimum of molecular models. The results in this study matched the current best in the field. The algorithm was written in C++.

Quality of Stochastic Versus Deterministic Algorithms for the Unweighted Minimum Vertex Cover Problem

Paper: <http://www.edmondchuc.com/minimum-vertex-cover-problem/paper.pdf>

Studied the minimum vertex cover problem and compared the performance of two different algorithms. The algorithm was written in C++.

Car Parking and Blind-Spot Monitoring System

Video demo: <https://www.youtube.com/watch?v=CBZHZsfnGhI&t=24s>

A Tiva project to create an active blind-spot monitoring system by using ultrasonic sensors with LED and buzzer feedback. Buzzer was set up using Pulse Width Modulation. Ultrasonic sensors detected distance by calculating the time taken between each pulse. Written in embedded C.

Web Chat Application

<https://edmondchuc.github.io/web-chat-app/>

Real-time web chat application built using Node.js, Express, Socket.IO, RxJS and Angular. Contained features such as admins, groups, channels and the sending of images over chat.

REFERENCES

Available upon request.