**Last updated:** March 27, 2021

**Original copy (in case broken):** `https://mullikine.github.io/cv/`

# 1 Shane Mulligan; BSc (Computer Science)


| Phone | Email | City |
|---|---|---|
| +64 21 1462 759 | mullikine@gmail.com | Dunedin, NZ |


**Platform Engineering · Prompt Engineering · Test Automation · Natural Language Processing**

## 1.1 Shane Mulligan [Cover Letter]

Dear Hiring Manager,

Please consider me for this role.

I'm passionate about developing tooling and automating software development processes and applying programming language design to models of natural language. I am currently developing a new kind of programming language for running software on GPT-3.

In 2020 I contracted for FMG in Australia developing CI/CD infrastructure their vehicle autonomy team on AWS. I enjoyed creating productivity tools for the developers, working with GitLab CI and AWS EC2. I have returned to New Zealand since and been involved with my friend's project in realistic avatars (URL: `https://takaheai.github.io/`). I've also been through the Facebook interview process for a platform engineering job, so have completed this primer `https://github.com/donnemartin/system-design-primer`.

In 2019 I automated a pipeline to scale a platform for automating pull requests to GitHub. The pipeline scripts I made touched on everything from GCP, kubernetes, GitHub and AWS and utilised shell, golang, python and common lisp.

In 2018 I studied Information Retrieval and Deep Learning at the University of Otago. I developed a GitHub search engine as my project.

From 2016 to 2017 I maintained CI infrastructure for Crown Equipment Corporation for their autonomous forklifts.

I really enjoy emacs, chatbots, NLP and lisp. I work entirely within a terminal environment I have been automating since 2011. I have 10+ years experience in the workforce as a linux-based software developer and tester. I can research, create quality documentation, design and automate experiments. Please take a look at my blog (`http://mullikine.github.io`) I am very capable and hard working.

Thank you for your consideration,

Shane Mulligan

## 1.2 Links

- Blog: http://mullikine.github.io - Daily Blog
- Not-for-profit: `https://github.com/semiosis`
- AI Blog: TakaheAI (GitHub: Takahe)
  Ultra-realistic avatars and chatbots for the enablement of humanity.
- NLP Blog: semiosis (GitHub: semiosis)
  Semiosis is building programming languages to keep computing textual.
- ML Blog: infogetics (GitHub: infogetics)
  Infogetics maps software for transforming one type of information into another
- GitHub, gists, Most active GitHub users in New Zealand

## 1.3 Past CVs

This is here to communicate my background but not my direction.

- CV and written reference - 2015 - Application for C++ Development

### 1.3.1 Presentations

1. *<2021-02-19 Fri>* `prompt-engineering` Part 1: Building the environment `https://mullikine.github.io/posts/creating-a-playground-for-gpt-3-in-emacs/`

   **nlp** **gpt-3** **emacs**

   This presentation is about building a programming environment for the new programming-paradigm of **prompt engineering**.

2. *<2020-10-06 Tue>* Creating standalone UI applications for CLI tools using emacs and docker `https://mullikine.github.io/posts/codecraft-creating-uis-for-cli-programs-with-docker-and-emacs/`

3. *<2018-04-30 Mon>* COSC431: The Case for Learned Index Structures `http://github.com/mullikine/cosc431-assignment-2-presentation/blob/master/presentation.org`

### 1.3.2 Relevant blog articles for Platform Engineering / DevOps

- `GCP`
  - Tremendous Task: Searching for code on GitHub with BigQuery and GHTorrent // Bodacious Blog
  - CodeCraft: Stacklet - Save Money on the Cloud // Bodacious Blog
- `NLP`
  - Generating poetry with GPT-2/
- Deployment and `IaC`
  - `https://mullikine.github.io/tags/iac/`
  - CodeCraft: Stacklet - Save Money on the Cloud // Bodacious Blog
  - Deployment with IaC and Octopus // Bodacious Blog/
  - Automatic, interactive handling of build files and IaC // Bodacious Blog
  - Review of 'Terraform code quality - CloudSkiff' // Bodacious Blog
  - Terraform // Bodacious Blog/

  - OS-agnostic program installation with Ansible // Bodacious Blog
  - Review of 'Ansible and Terraform: Better Together' // Bodacious Blog
  - HashiCorp: Terraform, Vault, Vagrant, Packer // Bodacious Blog/
  - Provisioning with IaC // Bodacious Blog/
  - Ansible // Bodacious Blog/
- AWS
  - Review of 'AWS RoboMaker - Amazon Web Services' // Bodacious Blog
  - Autoscaling GitLab Runner on AWS // Bodacious Blog

## 1.4 Work experience

### 1.4.1 Fortescue Metals Group (contract)

**Job title** Senior Software Tester and Tooling

**Description** Software testing, CI/CD and deployment for autonomous mining vehicles.

**Job Application** Application to Fortescue Metals Group // Bodacious Blog

**Started** Jan 2020

**Ended** May 2020

| Area | Task |
|---|---|
| Point of failure analysis | FMEA |
| CI/CD | GitLab Runner, Docker |
| Tooling | ROS2, C++ |
| Deployment | Shell, Octopus |
| Platform Engineering | EC2, Terraform |
| Research | AWS RoboMaker |

### 1.4.2 CodeLingo Ltd

**Job title** Software Engineer (contractor)

**Description** CI/CD for autonomous GitHub pull requests.

**Started** Sep 2018

**Ended** May 2019

CI/CD, Platform Engineering with kubernetes, GCP and AWS. Development in many (10+) languages. CodeLingo is a SAAS for linting and automating pull requests on GitHub using a domain-specific language.

1. Examples of pull requests generated and made automatically

   These pull requests were generated by the automated CodeLingo platform.

   `https://gist.github.com/1860bfea2a9e1e3b3bbb96b95a11bdd0`

2. Languages used

| language | context |
|---|---|
| Golang | Platform Development. Tooling. Specifically, lexicons. |
| Shell | Deployment Pipeline |
| Ruby, Python, Go | GitHub API |
| ElasticSearch Lucene | Debugging |
| BigQuery standard SQL | GHTorrent / Bigquery for Github |
| JavaScript | Unit testing |
| emacs lisp | Building an environment for a new language (CLQL) |
| YAML, Go, Python, Ruby | Researching best practices and generating code |

3. Highlights

    - 0 to 1000 automated github pull requests over 8 months.
    - Pipeline outreach scaled the platform from 10 to 300 app installs.

#### 1.4.3 Crown Equipment Corporation

**Job title** Software Engineer

**Description** CI and testing for autonomous forklifts.

**Started** Jan 2015

**Ended** Sep 2017 (2 years 8 months)

HIL (hardware-in-the-loop) testing

CI/CD

C++ application development

Python unit testing

1. primary languages used

    - C++ 13
    - python

2. Responsibilities

    - HIL (hardware-in-the-loop) rigs / integration tests
    - Implement driver for 3D Basler Camera

3. Highlights

    - Built a platform for correlating error messages with logs and code using the the Sphinx open source search engine.

#### 1.4.4 TracMap

**Job title** Software Engineer

**Description** Embedded software development for GPS navigation.

**Started** Jan 2011

**Ended** Mar 2013 (2 Years 2 months)

Embedded C++ application development. Porting Firmware.

1. primary languages used
    - C++
    - Python
    - JavaScript
    - PostgreSQL
2. Highlights
    - Ported the firmware from the older TM4 head units to the then prototype TM5 headunit.
3. Images
    - 2012: TM4 needs love
    - 2019: First TM5 returns home

## 1.5 Open-source Projects

### 1.5.1 Age of Kings Trigger Studio

`http://aok.heavengames.com/blacksmith/showfile.php?fileid=12103`

The most popular Age of Empires II scenario editor. Downloaded 11,000 times since 2014. Used in making campaigns for Age of Empires II HD edition and expansions.

| Software | Purpose |
|---|---|
| IDA Pro | Reverse engineering |
| Visual Studio 2005 | Compiling |
| emacs/vim | Programming |

## 1.6 Skill set

- Natural Language Processing
- Prompt Engineering
- emacs
- Language agnostic / polyglot
- Self-management
- Continuous learning
- Automation
    - Build tools, pipelines, cloud.
- Functional Programming

## 1.7 Current projects

### 1.7.1 `pen.el` - Prompt engineering in emacs

Facilitates the creation, development, discovery and usage of prompts.

Prompt engineering is a new programming paradigm based on querying large Language Models such as Google's BERT and GPT-3.

- `https://github.com/semiosis/pen.el`
- Autocompleting anything with GPT-3 in emacs // Bodacious Blog

Create elisp functions based on GPT-3 prompts. Chain GPT-3 queries together using keyboard macros and functions Interactively query, generate and transfrom both prose and code

Use GPT-3 as a search engine within emacs.

- Search the internet
- Search documents

#### 1.7.2 `examplary` - An example-oriented DSL that can be used to construct and compose NLP tasks

This is an extension of `pen.el`. One uses this DSL to create more advanced prompts.

`https://github.com/mullikine/examplary`

#### 1.7.3 LSP Server in Clojure

I have an ongoing project which is my expansive glossary. I'm currently building an LSP server in clojure which highlights any editor with things that I know or want to incorporate into my lexicon. I use this glossary system as the scaffolding to apply NLP algorithms and to help me to learn things. This is intrinsic and present within everything I do on my computer.

Updates and demonstrations of the glossary system // Bodacious Blog

### 1.8 Education

| Degree | Field | Insitution | Status |
|---|---|---|---|
| BSc | Computer Science | University of Otago | Finished 2010 |
| PGDip | Information Retrieval | University of Otago | Started 2018 |

—

I have also been meticulously going through Ryan Ong's NLP 365 series, which he created as he went through his Masters and PhD.

A tour of Ryan Ong's - NLP 365 // Bodacious Blog

#### 1.8.1 Highlights

- Won the COSC343 robot competition
- Completed a GitHub search engine using GHTorrent and BigQuery.

#### 1.8.2 Interest papers

- Health science 1st year
- Chemistry (200 level)
- Anatomy (200 level)
- Information Retrieval (400 level)
- Neural Networks (400 level)

#### 1.8.3 Books

- Read:
  - Neural Networks From Scratch (SentDex)
  - Introduction to Information Retrieval - Manning, Christopher D.
  - Language Modeling - Stanford University
  - Boolean Retrieval - Cambridge

  - Myst: The Book of Atrus - David Wingrove, Rand Miller, and Robyn Miller
  - The Elements of Eloquence - Forsyth, Mark
  - Coherence in natural language. Data structures and applications.

- Reading:
  - Search engines information retrieval in practice - Croft, W. Bruce Metzler, Donald Strohman, Trevor
  - HaskellBook - Christopher ALlen, Julie Moronuki
  - 1909.05858 CTRL: A CONDITIONAL TRANSFORMER LANGUAGE MODEL FOR CONTROLLABLE GENERATION - Salesforce
  - Using Search-Logs to Improve Query Tagging - research.google.com
  - 2101.03961 - switch transformers - Google Brain

—

I convert my books to text so I can do NLP on them and build up my glossaries.
`https://mullikine.github.io/posts/calibredb-inside-emacs-with-text-pdfs/`

## 1.9 Volunteer work

### 1.9.1 School of Computer Science

Ongoing. Tutoring junior school and high school students.

**started** 2018

**ended** Dec 2019

1. proud moments
   (a) portfolio gallery (circa. 2018) Guided 2 high school students in building an interactive portfolio gallery for the Otago Settler's Museum.
      - Taught CSS, javascript, php.
      - Used Google Cloud NLP for transcribing spoken queries

   (b) built a bot that can play the board game codenames (circa. 2018)
      - Taught the concept of 'word vectors'.

### 1.9.2 Experience

| | Company | Context | |
|---|---|---|---|
| **Containerization** | FMG | Docker, docker compose, docker swarm | 5 months (2020) |
| **Orchestration** | CodeLingo | Making tooling for K8s, EKS, minikube | 8 months (2018, 2019) |
| **Software deployment**, **AWS** | FMG | AWS, GitLab CICD, Octopus. | 5 months (2020) |
| **GitLab CI/CD tools** | FMG | Automating Build and Deployment | 5 months (2020) |
| **AWS** | CodeLingo | Platform Automation CodeLingo | 8 months (2018, 2019) |
| **GCP** | CodeLingo | Searching for github code | 8 months (2018, 2019) |
| **Golang** | CodeLingo | Go application development | 8 months (2018, 2019) |
| **Test automation** (**Hardware**) | Crown | Localisation, smoke, integration tests. | 2 years (2016, 2017) |
| **Test automation** (**Hardware**) | TracMap | Prototype traceability matrix | 2011 |
| **C++ testing** | TracMap | C++ application testing | 2011 |
| **Test automation** (**JavaScript**) | TracMap | SAAS testing | 2012 |
| **Python testing** | TracMap | SAAS testing | 2013 |
| **C++** | Crown | C++ application development | 2015, 2016 |
| **C++** | TracMap | C++ application development | 2011, 2012 |
| **Python testing** | Crown | Selenium, HIL | 2016, 2017 |
| **Test automation** (**Golang**) | CodeLingo | | 2018 |
| **Point of failure analysis** | Crown | Test logs | 2016 |
| **Web Development** | | | 2005-2020 |

**Evidence**

| Area | Evidence from my blog |
|---|---|
| Using OpenAI in my development | openai // Bodacious Blog |
| Language Server Protocol | Creating an LSP mode for racket // Bodacious Blog |
| Language Development | An example-oriented DSL that can be used for NLP tasks. |
| Templates and Diagrams | Templating mermaid diagrams // Bodacious Blog |
| Code Generation | Translating Haskell to Clojure with GPT-3 // Bodacious Blog |
| ChatBot / GPT-2, GPT-3 | TakaheAI An operating system based on GPT-3 // Bodacious Blog |
| Natural language processing | Suggesting words with KeyBERT and pytextrank spaCy in emacs |
| Automating terminal applications | Automating rat, a powerful productivity tool // Bodacious Blog |
| Build tool automation | Automating build systems for many languages |
| Polyglot programming | Languages supported by my development environment |
| Scripting / pipelines / automation | Reading YouTube rather than watching it |
| Building terminal user interfaces | Complex Dwarf Fortress macros with tcl/expect, emacs and tmux |
| Building debugging tools | tooling TensorFlow Debugger (tfdb) and emacs DAP mode for emacs |
| Metaprogramming | Practical macros in Racket C++ template metaprogramming with Racket |
| Preprocessing | Filtering text streams |
| Algorithms | The Illustrated Transformer `https://mullikine.github.io/glossary.html` |
| Information retrieval | Tremendous Task: Searching for code on GitHub with BigQuery and GHTorre |
| Documentation | Graphviz and Hugo Entropy, Cross-Entropy and KL-Divergence |
| Web development | The Semantic Web and Ontology |
| Deep learning | The Illustrated Transformer |
| Continuous Integration | `http://codelingo.io` Overview of modern Continuous Integration tools |
| Functional programming | GHCi and Haskell code intermixed in babel |
| Telco / Microservices / Kubernetes | telco github GCP |
| Bayes | (WIP) Probabilistic programming with problog |
| Building editing environment | Browsing sqlite3 databases with edbi for emacs |
| Research / learning | Review of 'Language Engineering; Harnessing the Power of Language (2004)' |
| Code refactoring and linting | CodeQL by GitHub and Semmle CodeLingo vs Linters |

I am always learning:
`https://mullikine.github.io/glossary.html`

### 1.9.3 Programming languages

I support many languages in my environment.
`https://mullikine.github.io/posts/emacs-languages-supported/`
—

**Language strengths**

| Language | Work Experience | Language | Work Experience |
|---|---|---|---|
| Python | yes | scheme / racket | |
| Clojure | yes | problog | yes |
| Shell (POSIX, bash, zsh) | yes | Perl | yes |
| c | yes | PerlRE and PCRE | yes |
| c++ (98) | yes | sed | yes |
| c++ (13) | yes | awk | yes |
| SQL (BigQuery) | yes | JavaScript | yes |
| Go | yes | Java | yes |
| Haskell | | JQ | yes |
| Common Lisp | | GraphViz | yes |
| TCL (expect) | yes | Latex | yes |
| emacs lisp | yes | VimScript | yes |
| Rosie RPL | | Prolog | |

| Cloud-Native Language | Work Experience |
|---|---|
| CLQL | yes |
| CodeQL (GitHub / LGTM) | |

## 1.10 Citizenship

- Australia
  Eligible for an E-3 visa in the USA.
- New Zealand

## 1.11 References

### 1.11.1 Dr Zhiyi Huang

**email** `zhuang@cs.otago.ac.nz`

Associate Professor
Department of Computer Science
University of Otago
Dunedin, New Zealand

### 1.11.2 Jesse Meek

**email** `waigani@gmail.com`

CEO
CodeLingo
Dunedin, New Zealand

## 1.12 Contact details

**phone** `+64 3 4777 071`

**mobile** `+64 21 146 2759`

**mobile** `+64 22 589 5536`

**email** `mullikine@gmail.com`

### 1.12.1 Linkedin

www.linkedin.com/in/shane-mulligan-811b942b/

## 1.13 Personality

**Mediator (`INFP`)** Retrospective on my personality // Bodacious Blog

## 1.14 Colophon

If there are missing pages or the formatting is off, you can find an original here:

**Online version:** `https://mullikine.github.io/cv/`

**PDF version:** `https://mullikine.github.io/ox-hugo/cv-newest.pdf`

**Last updated:** March 27, 2021