

Brian May

Resume: Appendix

Organisations

- Elixir** Elixir Melbourne is the meetup group for Elixir in Melbourne.
- FSM** Free Software Melbourne is a group that promotes usage of free or open source software. They have regular meeting and a mailing list.
- MelbournePUG** Melbourne Python Users Group is the meetup group for Python in Melbourne. They have regular meeting and a mailing list.
- LA** Linux Australia promotes the use of open source software through Australia. I regularly attend the linux.conf.au. LA is also responsible for the PyConAu conferences, which I attend.
- LUV** Linux Users Victoria is a Linux user group that discusses Linux issues.
- PTUA** Public Transport Users Association: a lobby group with the aim to improve and promote the use of public transport as an alternative to being dependent on the car. They have regular meeting and a mailing list.

Projects

Many of my skills come from open source projects. These are available on github.

Various web services I have accounts on:

- Homepage - <https://linuxpenguins.xyz/brian/>
- Stack Overflow - <http://careers.stackoverflow.com/penguinbrian>
- Github - <https://github.com/brianmay/>
- GitLab - https://gitlab.com/u/penguin_brian
- LinkedIn - <https://www.linkedin.com/in/brian-may-44824521>

The following is a list of some of my projects I have been involved in, both open source and proprietary.

2020 to present

- BUPA** Technologies: Python, Django, MySQL
Employer: WSP Digital
Client: BUPA
Upgrade legacy project that uses Python 2.7, Django 1.9 to use Python 3.8,

Django 2.2, and Docker containers for deploys.

2019 to present

- PhoneDB** Technologies: Elixir, Phoenix, PostgreSQL
An application for recording phone numbers of incoming calls from Freeswitch for automatic call filtering.
In writing this code, I learnt about Phoenix live view, a technology that allows the creation of interactive and responsive web apps entirely in Elixir.
- Design Manager** Technologies: Microsoft Access
Employer: WSP Digital
Generate quote for project to convert database to Python/Django.
- Geotechnical Digitisation** Technologies: Python, Django, PostgreSQL
Employer: WSP Digital
A Django/Python/Geos application for recording details of boreholes.
I am responsible for updating the backend code to as per requirements for the front end.
- Energy Rating** Technologies: Python, Django, PostgreSQL
Employer: WSP Digital
An established application for storage and calculation of data used for energy efficiency star ratings used throughout Australia and NZ.
I am involved in support work for solving day to day issues that user's face as well as work to keep the code up to date with the latest legislation.

2018 to present

- TeNerves** Technologies: Elixir, Geos, Nerves, PostgreSQL, Robotica
A Nerves/Elixir based tool for automatically monitoring our Tesla and making warnings via Robotica to remind us to do things like plug in the car overnight for charging.
In writing this code, I learnt about using Ecto, a functional library for talking to an SQL database, from within a Nerves application.
- lifx** Technologies: Elixir, LIFX
Elixir library to control LIFX lights, used by Robotica.
I forked this code from an existing repository, and learnt how to write code that does reliable sending of messages with retries from an Elixir GenServer.
- ex_tesla** Technologies: Elixir, Tesla, Geos
Elixir library for talking to the Tesla API for cars.
- Robotica** Technologies: Elixir, Nerves, MQTT
A rewrite of the Python based tool using nerves/Elixir. This is a tool for

management of events around my house on several Raspberry Pis.

In writing this project I learnt a lot about Nerves and how to create custom images for Nerves deployments.

MCA	Technologies: Elixir, React An Elixir/React application for working out train connections using real time data from PTV.
Time tracker	Python application for recording time sheets in YAML format.
ICMP	Technologies: Python, Django, PostgreSQL Employer: WSP Digital Congestion control and management.
Macquarie Dictionary	Technologies: Python, Django, PostgreSQL Employer: WSP Digital Macmillan Publishers Australia
State-wide electronic speed limit data	Technologies: Python, Django, Android, IOS Employer: WSP Digital Client: QLD government Generate BID for tender.
AIMES	Technologies: Python, Django, PostgreSQL Employer: WSP Digital IOT Smart sensors for transport.
Barrier Asset Management System	Employer: WSP Digital Client: VicRoads Generate BID for tender.
Inventory Management	Technologies: Python, Django, PostgreSQL Employer: WSP Digital Client: BlueScope Inventory management system using Django/Python.
DaRT	Technologies: Python, Elixir, Pandas, ElasticSearch, AWS Employer: WSP Digital Client: NSW Trains Python/Elixir code to analyse and join multiple tables of different formats from multiple sources in order to generate statistics on trains such as dwell times and run times. In writing this code I learnt about joining tables from different sources and how to do so in a fast and efficient manner.
Design Package Workflow	Technologies: Python, Django, PostgreSQL, JIRA Employer: WSP Digital Client: SPA Python/Django Code to import data from legacy PM system into Jira.

2017 to present

- Remote** Technologies: ESP32, MicroPython, AsyncIO
A MicroPython based 3d printed remote control for Robotica.
- aiolifxc** Technologies: Python, LIFX
Python library to control LIFX lights, used by Robotica.
I forked this code from an existing repository and adapted it to my requirements.
- Robotica** Technologies: Python, AsyncIO, MQTT
A Python based tool for management of events around my house on several Raspberry Pis.
- spud-client** Technologies: Typescript, Angular2
A front-end for the spud photo album.

2016 to present

- MyRocc** Technologies: Python, Django, Geos, PostgreSQL, Cumulocity, ECS, Shippable
Employer: WSP Digital
Client: VicRoads
Website for management of IOT devices along freeways.
I contributed to a new greenfield development as a team using Agile management.
- Online Forms** Technologies: Python, Django, PostgreSQL
Employer: WSP Digital
Client: FACS
Python/Django code for generic form based system.
I successfully contributed to a large complicated code base in a large team effort.
- Debian LTS** Technologies: Mostly C Employer: Freexian
Freexian sponsors creating security updated packages for legacy Debian based operating systems. This is a task that must be done very carefully so as not to break critical computer systems. It involves coordination with other team members, the Debian security team, the package maintainers, upstream authors, and any other relevant party.

2012

- TLLDAP** Technologies: Python, LDAP
A Python library for accessing LDAP servers in a server/schema independent manner. Used by Placard and Karaage.

2010 to present

- Placard** Technologies: Python, Django, LDAP

Employer: VPAC

A Django based python application for management of LDAP.

Karaage

Technologies: Python, Django, LDAP, MySQL, Slurm, AAF Shibboleth
Employer: VPAC → UniMelb

A Django based python application for management of users on their super computer clusters using LDAP.

In working on this project I rewrite legacy code in such a way to minimize downtime and increasing the reliability of the code.

2000 to present

Spud

Technologies: Python, Django, PostgreSQL

A photo database using Django/Python for the REST back-end and Angular2 for the front-end.

This project has been valuable experience in learning how to efficiently store tree like structures in a SQL based backend. It has also enabled me to understand writing the client as a JavaScript based web app.

Lintory

Technologies: Python, Django, MySQL
Employer: VPAC

An asset management system, using Django/Python.

Sitebuilder

Technologies: PHP, MySQL, PostgreSQL
Employer: IVT

System management tool, using PHP and sudo calls.

1990-2000

Radio test - Motorola

Technologies: Windows 3, Fortran, Borland C++
Employer: Motorola

I designed and ported a legacy Fortran application to C++.

Debian packaging

Technologies: variety

Packaging packages for inclusion in Debian.

Debian is a free non-proprietary operating system, which is designed and implemented by volunteers world wide. I maintain a number of packages for Debian and I am part of the Debian Python Modules Team. I contribute to mailing list discussions on the future directions that the operations should take.

Good communication is absolutely vital. All communications must be done via electronic means over different time zones. If there is a misunderstanding, it is not possible to meet face to face to resolve the problem. As such, I have obtained skills required for communicating to other Debian developers in an effective manner, using tools like Email and the Debian bug tracking system.