

# Andy Bulka

Melbourne, Australia  
abulka@gmail.com  
<http://www.andypatterns.com>  
<https://github.com/abulka>

*Hi, my name is Andy Bulka and I am a Python/Django and Web/Nodejs/Electron software developer. I like to think about how to better understand complex software, and the role that visualisation technologies (diagrams) can play in making programmers more effective.*

---

## Technical Skills

**Likes:** python html node.js design-patterns oop agile django javascript flask  
**Dislikes:** php objective-c c++

---

## Experience

*Jan 2017 → Current* **Director of Software Development** – [Wware Consulting](#)  
python, node.js, electron, javascript, html, django, flutter

Python/Django, Flutter and Website Software Development Contracting.

Desktop App development in wxPython and Electron.

Creation of various products GitUML, Pynsource etc.

---

*Oct 2014 → Mar 2016* **Senior Software Developer, Development Lead (Python, Django)** – [Telstra](#)  
python, django

Technical lead of team of 6 – including offshore. Liase with PM and BAs. Learnt and maintained a substantial existing code base written in Django and the Django REST framework.

---

*Mar 2013 → Sep 2014* **Senior Software Developer** – [Nexus6](#)  
python, javascript

Looked after 2 large web sites built in Python. SQLAlchemy database ORM with Postgres backend. Celery for background tasks, managed and enhanced various job system queues e.g. image thumbnailing, background processing of csv imports etc.

---

*2009 → Mar 2012* **Director of Software Development** – Magian Media Studio  
actionscript, html, ios, ruby-on-rails

Looking after a team of approx. eight developers working mainly in Flex, Ruby and Delphi - developing world class touch screen interactives and soundscapes for museums, science exhibits and art installations around the world.

---

*Mar 2005 → Nov 2008* **Technical Director & Chief Software Architect** – [Austhink Software](#)  
.net, uml, agile

Served on executive team, helped build a multi-million dollar company. Designed and architected two world class software products. Supervised PhD and related university teams in fundamental research used in the product. Gathered requirements, wrote functional specs, led design discussions, designed and diagrammed architectures, supervised the building of Rationale and bCisive, Licensing System – to support sales and key licensing for businesses and schools + Academic Papers – several academic papers produced as a result of graph layout research.

---

## Education

*Jan 1980 → Oct 1985* **B.A. Computer Science, Philosophy (Hons)** – University of Melbourne  
python, ruby, .net, design-patterns, html

Software Architecture, Software Design, Technical Director, Project Management. Experience in systems built in C# and .NET 2.0, Python and Delphi. Specialise in complete architectures for highly visual applications.

Specialties Expert in Software Design Patterns. Regularly speaks at user groups and occasionally trains others in Design Pattern workshops.

---

## Projects & Interests

*Jun 2021 → Current* **abulka/generator-electron-flask** – <https://github.com/abulka/generator-electron-flask>

Yoeman project generator to create an Electron app running a Flask server inside itself. Deployable via PyInstaller and Electron Forge packager.

---

Jan 2010 → Current **Pynsource (Open Source, GitHub)** – <https://github.com/abulka/pynsource>  
python, wxpython

UML tool for python source code.

Pynsource UML tool - reverse engineer Python into UML class diagrams Major new release now parses Python 3. Has zoom, layout, ASCII UML and PlantUML rendering support. Pynsource is also, as far as I know, the only UML tool that recognises Python instance attributes (not just class attributes). This means that expressions like `self.myattr` will result in a proper attribute "myattr" in the resulting UML class. Ready to run binaries are available for Mac, Windows, Ubuntu 18 and 16, Snap - as well as an open source Github repo. Community edition free and open source.

---

## Public Artifacts

May 2020 **TodoMVC implemented using a game architecture— ECS.** – <https://medium.com/@abulka/todomvc-implemented-using-a-game-architecture-ecs-88bb86ea5e98>  
javascript, entity, components, game-engine

Building traditional GUIs with the Entity Component System

Mar 2020 **TodoMVC implemented using traditional OO, Controllers and Events.** – <https://medium.com/@abulka/todomvc-implemented-using-traditional-oo-controllers-and-events-5e4c09f80cd4>  
javascript

TodoMVC implemented in a classic Object Oriented way

Apr 2019 **Getting Python and wxPython apps into the Ubuntu app store** – <https://medium.com/@abulka/getting-python-and-wxpython-apps-into-the-ubuntu-app-store-ccca7ae537a3>  
python, ubuntu

Have you ever dreamt of getting your own apps in the Ubuntu app store? Here's how.

Getting Python and wxPython apps into the Ubuntu app store Have you ever dreamt of getting your own apps in the Ubuntu app store? Here's how.

Jan 2019 **Building a deployable Python-Electron App** – <https://medium.com/@abulka/electron-python-4e8c807bfa5e>  
python, electron, windows, macos, linux

Building a normal, deployable application for Mac or Windows, using Python 3 is hard. There is nothing like Visual Studio or Delphi...

Building a deployable Python-Electron App Article on Medium - Using Electron as a GUI toolkit for Python. Argues that in a sense, Google, Apple etc. and thousands of Javascript library developers are working on improving the browser GUI toolkit every day, every hour vs. a few developers working on old toolkits from the 90's. In depth article shows how to use Python 3 inside Electron and deploy your GUI app using PyInstaller.

Jan 2019 **Django View logging is back-the-front** – <https://medium.com/@abulka/django-view-logging-is-back-the-front-7f9701d501de>  
django, python

Have you ever looked through your Django log files and console messages, felt something was out of order and didn't make sense, but...

Jan 2018 **Async/await for wxPython** – <https://medium.com/@abulka/async-await-for-wxpython-c78c667e0872>  
python, wxpython

Python 3 GUI apps with asynchronous functionality

Async/await for wxPython Article on Medium - Python 3 GUI apps with asynchronous functionality. The wxPython GUI toolkit is arguably important because it's free, even for commercial purposes (unlike PyQt). Article even contains a slightly controversial section entitled "Is async/await an anti-pattern?" which refers to the phenomenon of the `async` keyword potentially spreading through one's codebase, and some thoughts on how to mitigate that.

Jan 2011 **AndyPatterns :: UML\_layout** – <http://www.andypatterns.com/index.php/blog/uml-layout/>

Developing a Layout Algorithm for UML diagrams

Apr 2009 **AndyPatterns :: Refactoring to PureMVC** – [http://www.andypatterns.com/index.php/blog/puremvc\\_refactor/](http://www.andypatterns.com/index.php/blog/puremvc_refactor/)  
model-view-controller, design-patterns

PureMVC - What's all the fuss about?

I have long been fascinated by the MVC (Model View Controller) architectural pattern, first conceived in the 1960's. What it promises, to those who can fathom its mysteries, is an orderly way of organising your application architecture. You define a de-coupled domain/business model that is oblivious to any GUI that might be displaying it. A mediating controller class usually looks after the dirty details of moving data between the model and GUI. You can even have multiple GUI representations of your single model e.g. a pie chart view and a bar chart view simultaneously displaying the same information out of the model.

---

Mar 2009 **AndyPatterns :: PureMVC - Architectural Pattern - Minimal wxPython Example** – [http://www.andypatterns.com/index.php/blog/puremvc\\_minimal\\_wxpython/](http://www.andypatterns.com/index.php/blog/puremvc_minimal_wxpython/)  
python, wxpython

This demo illustrates the simplest way to build a PureMVC application in python using wxPython as the GUI. A string in the model is displayed in the GUI and anything you type gets...

---

Jan 2008 **AndyPatterns :: Easy Dependency Injection and Mocking in Ruby** – <http://www.andypatterns.com/index.php/blog/dependency-injection-in-ruby/>

You don't need fancy dependency injection frameworks to replace methods in a dynamic language like Ruby or Python.

---

Jul 2005 **AndyPatterns :: Object Relational Mapping Pattern - using SQLAlchemy** – [http://www.andypatterns.com/index.php/blog/object\\_relational\\_mapping\\_pattern\\_using\\_sqlalchemy/](http://www.andypatterns.com/index.php/blog/object_relational_mapping_pattern_using_sqlalchemy/)  
python, sql

Relational databases and OO memory models use different paradigms. Mapping between one and the other is fiddly work so an auto-mapping tool like Sql Object (python framework) can...

---

## Apps & Software

Sep 2021 **Toolback**  
flask, javascript

A modern take on 'Hypercard' and 'Toolbook' - instant productivity in the modern era Drag and drop components - all of which can have scripting added Javascript programming - add scripts to components Add more javascript at the page or app level Easy event binding Call methods on other components using events F3 key instant switch between design mode and run mode

---

Sep 2020 **Brew Moments** – <https://abulka.github.io/brew-moments/>  
flutter, android

This brew calculator uses fast sliders and up down arrows to tune in your brew. Enter in the amount of coffee or amount of liquid and a brew ratio e.g. 1:16 and the other variables will be calculated for you.

The brew ratio calculator has the ability to save brew ratios as "moments". These saved moments can have notes, grind, water temperature and notes for recreating your favourite brews.

---

Jan 2019 **GitUML** – <http://www.gituml.com>  
django

Generate UML diagrams from Python code stored in GitHub repositories. Diagrams automatically update when you push code using git. Browse through community UML diagrams, surf through awesome listed GitHub repositories and visualise the architectures and patterns in them. Simple, point and click interface to reverse engineer into UML - add PlantUML markup customisation and HTML annotations.

---

Jan 2018 **Pynsource (Pro)** – <http://www.pynsource.com>  
python, wxpython, wxwidgets

Pynsource UML tool - reverse engineer Python into UML class diagrams Major new release now parses Python 3. Has zoom, layout, ASCII UML and PlantUML rendering support. Pynsource is also, as far as I know, the only UML tool that recognises Python instance attributes (not just class attributes). This means that expressions like self.myattr will result in a proper attribute "myattr" in the resulting UML class. Ready to run binaries are available for Mac, Windows, Ubuntu 18 and 16, Snap - as well as an open source Github repo. Community edition free and open source.

---

Jan 2018 **Python to RPN** – <https://pyrpn.atug.com/>  
python, calculator, hp

Run Python 3 on vintage HP calculators!

Run Python code on HP calculators from the 80's using this Python to RPN code converter. Convert Python 3 (subset) into runnable RPN via this online website. The HP42S is considered by many to be the best calculator Hewlett Packard ever made. It is one of the first popular programmable devices - just as PC's were just beginning. You programmed it in a language called RPN. We are talking about the early 1980's here folks. Python to RPN is a new way to program the vintage HP calculators like the HP42S, the reborn Swiss made DM42, and the Free42 emulator on desktop, iOS and Android.

---

Jan 2017 **Print42** – <https://www.print42.atug.com/>

Print42 is a log tailing GUI program which optionally echoes log lines to thermal printer tape. Electron-Python app, non-invasive log file annotation tool that supports thermal printers. Its like a modern thermal printer version of a ticker tape for tailing log files. It features display font size options, search and filtering. You can also annotate the output with text fragments that you type in - or screenshots that you paste in.

---

Jan 2000 **AndyPatterns :: Blog** - <http://www.andypatterns.com/>  
.net, python

My Blog - my articles and software apps

---

## Tools

**Favorite Editor:** Vim, Vscodc, Pycharm