

James Cooke

London, United Kingdom
hi@jamescooke.info
<http://jamescooke.info>

For the last 14 years, I've been working with internet in some form or another, primarily web. Daily I use automated testing and deployment to generate best value for clients. My goal is to work with businesses that are iterating and improving their products over a long-term life cycle, delivering measurable value to their customers.

Technical Skills

Likes: python django git linux bash vim postgresql amazon-redshift elasticsearch tdd java

Experience

Python Developer – Coding Cooke Ltd
python

Dec 2013 → Current

At the end of 2013 I started working as a Python contractor with my primary goal to work with multiple clients and gain as much experience of various technical environments as possible. I have been able to work on a wide variety of projects, some of the most recent ones are highlighted below.

My focus at Coding Cooke is to create high quality, automatically tested code that can be maintained after any particular contract has ended. I often train my clients' development teams to work with, maintain and improve on that code going forwards.

Coding Cooke Contractor – UKTI
python, dynamics-crm-2011, git, django

Jun 2016 → Aug 2016

UKTI are developing an internal tool called Data Hub to enable them and their partners to better understand, manage and record their relationships with businesses. I worked on building an integration test suite for the Python Microsoft Dynamics 2011 library, as well as conducting some research into the existing data and technical needs of the project.

Coding Cooke Contractor – Huq
python, postgresql, django-rest-framework, ansible, git

Nov 2015 → Apr 2016

Huq is an analytics provider for mobile app developers. As the sole backend developer on this project, I worked on the three main parts of the backend system to get them production ready for this start-up's launch:

- Collector: To provide a quick turn around and collect all data on Huq's API.
- Resolver: To correlate location information provided by geolocation services into concrete place data.
- Reporting: The API that provides aggregate information to Huq's customer dashboard.

Coding Cooke Contractor – Construct
python, mariadb, django, django-rest-framework, apple-push-notifications, git, couchbase

Jan 2016 → Apr 2016

Previously I had worked with Construct to build their API server, running Django on top of MariaDB and Dropbox datastore to provide account management, synchronisation and notification services for a project management iOS app used by the construction industry.

In light of Dropbox turning off their Datastore in April 2016, I worked on a migration of the server code and accounts to run using Couchbase SyncGateway.

Coding Cooke Contractor – Scoota
amazon-redshift, python, git, django, django-rest-framework, grafana

Apr 2015 → Oct 2015

Built on Amazon's Redshift data warehouse, I designed Scoota's current warehouse relational structure, data ingestion and analysis processes.

Millions of events are analysed daily, at regular intervals and in parallel, and compiled to per-Campaign report tables which can be queried by Django views that render charts in D3. I've most recently developed a new integration test suite and dynamic schema change system that allows the client developers to upgrade the warehouse without taking it offline.

Coding Cooke Contractor – Decoded
salesforce, python, django, tdd, git

Jan 2015 → Apr 2015

Migration of this training company's business data off Salesforce to a new API server that now lives at their heart of their internal management and external client app ecosystem.

I built migration rules and validation for the courses, contacts, invoices, locations and facilitators to ensure that my client's business data was validated and a new solid database is created. I worked with the client developers to improve their Django and Python, teach them TDD and allow them to maintain and improve the system after deployment.

Lead developer – Quib.ly
pyramid, python, postgresql, sqlalchemy, solr, redis, chef, git

Nov 2012 → Nov 2013

Responsible for the backend Python code, new feature project planning, server maintenance and release deployment for this Pyramid-driven Q&A website for parents. Since taking over the back-end development of the project, I've managed the site through its public launch, Techcrunch and BBC traffic spikes, and continued development releases.

During my time at Quib.ly, I've implemented a TDD structure for backend development which has raised test coverage from 32% to 75%, the primary side-effect of which has been to improve core code quality and team sanity. Along with Python and PostgreSQL, the stack includes some Solr and Redis, and can all be deployed from scratch with Chef Solo.

Director – Fublo Ltd
php, python, django, git, wordpress, solr

Dec 2010 → Sep 2012

Working through the entire business process for developing websites for external clients and running a small agency of three from our London studio. From initial meeting and requirements capture, to proposal writing, project planning, backend development, project management and ongoing advice and support. Initially most development was based on WordPress with Fublo providing plugins and themes for clients. However in 2012 we moved to use Django primarily for client work with great results. Clients included Fallon London, We Are Social, Time Out and NeuxPower.

Freelance digital consultant – James Cooke Freelance
php, wordpress, solr

Jan 2010 → Dec 2010

2010 was the freelance precursor to Fublo. Mark (Fublo co-founder) and I worked as freelancers from a small office on primarily WordPress based projects for clients like Virgin and Time Out London.

Online Manager – Time Out International Ltd
google-analytics, adobe-analytics, ms-office, php

Jul 2007 → Dec 2009

Working within the licensing side of Time Out's business, I primarily assisted licensees with the creation and management of their Time Out websites in various territories.

Responsible for distilling best practice for optimal visitor behaviours and site usability and sharing that with the international teams and developing reporting systems for monitoring of revenue generation and performance of all Time Out sites in the network.

Web Developer – Time Out Communications Ltd
php, cvs, svn, apache2

Jan 2007 → Jun 2007

Working on the PHP codebase of Time Out London's website as part of the London development team. Some key projects included work on migration of the TO code base from php4 to php5, migration of the dev team from CVS to SVN and development of a new film site.

Web Services Developer – Bandwagon
php, svn, html, soap

Jul 2006 → Dec 2006

Bandwagon was a startup building a social-networking music community that to offer labels and unsigned artists a system through which they can promote, interact, and offer digital and mobile music downloads to the mass market of fans.

Day to day, I worked on the back-end of the website application to extend the offerings to users and improve the site's existing code.

Founder/Director – Root Seven Ltd
php, html, windows-2000

Oct 2004 → Jul 2006

Executive position involving the day to day management of the company, its employees, projects and growth. Technical roles include web strategy, application design, client and project management and project specification.

IT & Design Consultant – jclTconsulting
php, html, wordpress, mysql, windows-2000

Sep 2002 → Oct 2004

Freelance consultant working to provide IT advice and implementations to SMEs, NGOs and charities where cost efficiency and a strong online presence, brand and message are most important.

Industrial Strategic Planning Trainee – IBM
java, jvm, c

Apr 2001 → Sep 2001

Trainee position in the Strategic Development Department, part of the Java Technology Centre, Hursley, investigating implementation possibilities of new Java Virtual Machine profiling technologies.

Education

MEng Computing – Imperial College London
haskell, java, mono, prolog, minix

1998 → 2002

Projects & Interests

Stack Overflow – <http://stackoverflow.com/users/1286705/jamesc>

2012/03 → Current

Written 33 answers. Active in ubuntu, django, elasticsearch and facebook.

factory_djoy – https://github.com/jamescooke/factory_djoy

May 2016 → Current

Factories for Django, creating valid instances every time

restapidocs – <https://github.com/jamescooke/restapidocs>

Jul 2015 → Current

Templates for documenting REST APIs

chattydotfiles – <https://github.com/jamescooke/chattydotfiles>

Jun 2014 → Current

A simple and verbose dotfiles manager in Python

droidstrap – <https://github.com/jamescooke/droidstrap>

Mar 2013 → Current

Simple Bootstrap based theme with Droid fonts for Pelican blogs

Public Artifacts

API Documentation and the Communication Illusion – <http://jamescooke.info/api-documentation-and-the-communication-illusion.html>

May 2016

How can documenting an API help us to communicate with our team and API consumer developers efficiently?

Git: To squash or not to squash? – <http://jamescooke.info/git-to-squash-or-not-to-squash.html>

Nov 2013

This is currently the most popular post on my blog... From a Twitter discussion about the pros and cons of squashing, this post digs into the use of relog for keeping detailed history.

vi-nature everywhere – <http://jamescooke.info/vi-nature-everywhere-lightning-talk.html>

Nov 2013

This was a fun lightning talk at vim-london meetup - I explore how we can use "vi-nature" modal keyboard interactions with all our software and not just our editor.

Readings

Test Driven Development (The Addison-Wesley Signature Series) – Kent Beck – <http://www.amazon.co.uk/Driven-Development-Addison-Wesley-Signature-Series/dp/0321146530>

This book helped me to formalise and improve the structure of how I work with tests. It's a great book for those starting out with testing and also the intermediate programmer that wants to explore more.

Refactoring: Improving the Design of Existing Code (Object Technology Series) – Martin Fowler, Kent Beck, John Brant, William Opdyke, Don Roberts – <http://www.amazon.co.uk/Refactoring-Improving-Design-Existing-Technology/dp/0201485672>

Two big takeaways from this book for me were:

- That refactoring depends on good tests that cover the code being refactored.
- That the structure of Smalltalk allows programmers to work without hammering away at a keyboard... Instead the Smalltalk browser offers many refactorisations via a mouse click.

Tools

First Computer: Basic Tutor Computer
Favorite Editor: vim