

John Christian
Software Developer
(510) 304-1550
opal6502@gmail.com
<http://www.linkedin.com/in/opal6502>

Professional Objective

Having over 7 years of experience in software development infrastructure, that is, release engineering and QA, I have a very in-depth knowledge of scripting languages and software systems.

I would now like to transition over to developing on the modern web, the technologies have matured to the point where it's more exciting than ever, with HTML5, CSS3, Javascript and Node, developing incredible experiences on the web has never been better.

Not only that, the burgeoning marketplace of video tutorials makes learning more fun and at least virtually social, so expanding my skillset to include a thorough knowledge of Javascript, React, and Node has been one of the funnest learning experiences I've had thus far!

I am looking to begin a new career as a freelance developer on the MERN stack, I am very passionate about modern web technology and building great things, and look forward to contributing to exciting projects as soon as possible!

Thanks so much for your time and consideration!

Technical Skills

- *Languages:*
 - Javascript, Python, Perl, bash, Regex — 7+ years experience
 - Data structures, algorithms, and a wide variety of libraries

- *Libraries & Frameworks*
 - Javascript — 2 years study:
 - DOM manipulation
 - React.js
 - Node.js/NPM
 - Babel/Webpack
 - Python — 5 years experience:
 - OOP, Socket programming, regular expressions, threading, TkInter
 - PostgreSQL/psycopg2
 - Some Django and Flask
 - Some Django ORM and SQLAlchemy ORM
 - Selenium web automation
 - Perl — 6 years experience:
 - IO::Socket, LWP, Test::More, Carp, POE, Event
 - Regular expressions, algorithms and data structures
 - MySQL, PostgreSQL databases
- *Software & Tools*
 - VS Code, Pycharm, Emacs source code editors
 - Git, Subversion, and CVS version control systems
 - Jenkins, Zutubi Pulse continuous integration tools
- *Databases*
 - PostgreSQL, MySQL, Oracle, MongoDB
- *Operating Systems*
 - Linux (Debian, RedHat, SuSE), Solaris, MacOS X (10.x), OpenBSD, FreeBSD.
 - Other Unix Platforms: SGI Altix, AIX, HP-UX, Tru64, IRIX.

Recent Coursework

- Vanilla Javascript:
 - Modern Javascript From The Beginning — Traversy Media
 - Comprehensive intro to Javascript including language features, prototype system, DOM manipulation.
 - Vanilla Javascript Playlist — Traversy Media

- 50 hours of fantastic Javascript tutorials and projects, including advanced UI components, and various small apps.
- The Javascript Bible — Bogdan Staschuk
 - A fantastic 70 hour look at the javascript language from fundamentals, higher order functions, and a look at Babel/Webpack
- Javascript Algorithms and Data Structures Masterclass — Colt Steele
 - An in-depth look at writing algorithms and data structures in Javascript, including sorting, linked lists, graphs, trees, heaps, Dijkstra's algorithm.
- HTML/CSS:
 - Modern HTML & CSS From The Beginning (Including Sass) — Traversy Media
 - A look at HTML fundamentals and CSS styling, covering CSS2 floating, flexbox, grid, media queries, responsive design and element positioning
 - Learn HTML5 and CSS3 From Scratch — FreeCodeCamp.org
 - An 11 hour course on HTML and CSS, covering fundamentals, CSS3 flexbox and grid
 - Advanced CSS and Sass Flexbox, Grid, Animations and More! — Jonas Schmedtmann
 - A great CSS course, involved creating your own grid system, advanced styling, unit systems, structuring large Sass projects, responsive design
- React.js:
 - MERN Stack Front To Back Full Stack React, Redux & Node — Traversy Media
 - A great comprehensive course with projects using a React/Redux frontend and Node/MongoDB REST API backend
 - Modern React with Redux — Stephen Grider
 - A fantastic course covering the React and Redux frameworks in depth, the routing system, contrasting Contexts vs. Redux, and more.
 - ReactJS Crash Course 2021 — Traversy Media
 - A great intro to React.js fundamentals with a project app
 - Learn React JS - Full Course for Beginners — Programming

with Mosh

- More React.js fundamentals, another course with another perspective
- Node.js:
 - Node.js Crash Course — Traversy Media
 - Intro to developing in the Node.js environment, with NPM
- Python
 - Learn Python Programming Masterclass — Tim Buchalka
 - Comprehensive Python programming course covering fundamentals, data structures, GUI programming with TkInter
 - The Complete Python & PostgreSQL Developer Course — Rob Percival
 - Great course featuring Python algorithms and data structures and PostgreSQL programming
 - Python and Django Full Stack Web Developer Bootcamp — Jose Portilla
 - Very thorough Python Django course including Models, Forms, Templates, some Javascript, and deployments
 - The Python Mega Course: Build 10 Real World Applications — Ardit Sulce
 - An interesting variety of projects using Python libraries, including image processing, OpenCV, Pandas, Jupiter Notebook, Geographic Mapping, MathPlotLib, Flask, Beautiful Soup

Employment History

- ***Note: I've been on medical disability since my last role, but am now feeling completely recovered and eager to get back working ASAP, and I'm very much willing to negotiate compensation considering my extended absence, thanks very much.***
- **Technicolor** - QA Engineer (Consulting) - 11/2012 - 5/2013
 - Wrote automated test libraries for interacting with REST APIs for a streaming media application, involving customer and administrative use cases.
 - Test case development was done using a pretty full featured library developed by Nightmedia. (<https://github.com/>

nightmedia)

- **Rackspace Hosting** - QA Automation Engineer (Consulting) - 02/2012 - 11/2012
 - Wrote automated test libraries for interacting with REST APIs for managing Cloud objects such as Servers, DNS, including a websocket client running in a separate thread, queueing up server responses to be passed to automated test cases.
 - Wrote a smoke test for a Cloud services REST API using python unittest2, first run with the Nose python test framework, then added multi-threaded support using Pytest, reducing runtime by about 1/10th of original synchronous time.
 - Wrote Selenium automated tests for cloud-based DNS record management and Server configuration.
 - Configured tests to work in a SCons/buildbot, then SCons/Jenkins based CI environment.
- **EMC - Greenplum data computing division** - Release Engineer (Consulting) - 02/2011 - 11/2011
 - Wrote tools in Python for Release Engineering and QA infrastructure often using Django, as well as RESTful/XMLRPC services provided by the Jira bug tracking system and the Zutubi Pulse Continuous Integration environment.
 - Contributed additional test framework libraries to be used by QA Engineers.
 - Maintained a build framework for a PostgreSQL-based database product, using Python and Shell scripts.
 - Administered a Zutubi Pulse build and test framework environment with builds across several platforms including RHEL, SuSe, Solaris, MacOS X, and Windows
 - Coordinated patch releases and release candidate qualifications for regular and customer-specific builds.
- **Yahoo!, Inc** - Sr. QA Test Development Engineer (Consulting) - 04/21/2010 - 12/31/2010
 - Served as automated test designer and developer for the GMS (Grid Management System) project, a cluster management system that controlled Hadoop MapReduce and Oozie proprietary workflows in addition to hardware/system metrics. The application consisted of a Tomcat RESTful application

service and an Apache/PHP web front end, in addition to Oracle and HBase data storage facilities.

- Contributions include:
 - Developed test suites for new features based on Product Management specifications for the GMS (Grid Management System) project.
 - Wrote Perl module libraries to allow test cases to be written quickly and efficiently in an Object Oriented format.
 - Wrote automated suites to interface with Hudson CI and a proprietary test result logging application, in a Test::More format.
 - Coordinated the GMS 2.0 production release, working with Operations, Dev and QA.
- **Altair Engineering / PBS GridWorks Division** - QA Engineer / Tools Developer - *6/07/2005 - 04/21/2009*
 - Developed new test outlines for new PBS GridWorks features, and wrote a distributed automated testing framework along with 3 other QA Engineers.
 - Contributions include:
 - Designed and developed distributed automated test framework from scratch using Perl, MySQL, and PHP.
 - Performed testing for customer-specific patch and feature releases, including LRZ, NASA Goddard, and Cray, created facsimile test environments in-house when possible, or negotiated use of customer test system time.
 - Participated in the troubleshooting of intermittent customer bugs, sometimes re-creating customer environments in-house, sometimes an iterative process with a close dialogue with customer support.
 - Audited new PBS features from use case/requirements phases to design document to release candidate testing.
 - Performed delegation for the execution of manual tests, as well as technical help/direction to Bangalore QA department.
 - Duties during a release cycle (typically occurring over 4-month cycles) consisted of auditing the requirements/design of up to 3 new features, as well as writing automation for completed test specifications of the previous cycle, often for 2-to-3 features.
- **Postini, Inc.** - Release Engineer / Tools Developer - *11/15/2003 -*

3/31/2005

- Wrote and maintained existing tools to facilitate releng operations, as well as several new ones.

Contributions include:

- Designed and developed a new distributed system for build and test task management across the in-house dedicated build and test system, in Perl using the POE framework, utilizing XML messaging over secure socket connections.
- Designed and developed a new framework for project branch tracking, using Perl and MySQL.
- Vastly improved the branch merging utility I'd written in my previous stint, integrating it with the project branch tracking utility and streamlining the web interface to provide much more process automation.
- Participated in the troubleshooting and modification of automated tests.
- Developed various web-based tools for the tracking of release processes.

- **Movius, Inc.** - Configuration Manager - 4/15/2001 - 4/20/2002

- Helped develop product release cycle. Wrote tools to facilitate. Served several software development teams, a total of 15 developers, building the proprietary Java/C++ code base into packaged releases with 3rd-party tools including JBoss, Samba, MySQL.

Contributions include:

- Developed installer with packaging/dependency strategy from scratch in Perl, deploying and configuring the entire product environment from a curses-based interface.
- Developed installer configuration strategy from scratch, including a macro language to allow developers to specify configuration format in a very intuitive way.
- Wrote unit tests with JUnit and perl, providing coverage of supported configurations in a multi-node test-bed environment.
- Wrote entire build automation framework from scratch, using SVN, shell, ant and gmake.
- Created cluster deployment, failover strategy and installer scripts, using IBM middleware.

- **Postini, Inc.** - Release Engineer / Tools Developer - 11/20/2000 -

2/3/2001

Maintained CVS source tree and release schedule which involved merges that needed to happen on a weekly basis, amongst 4 or more active project branches being actively used by 15 developers.

Contributions include:

- Created first revision of a collaborative merge utility that would later be greatly improved upon during my second stint at Postini.
 - Created web-based tools for project branch tracking, in Perl and MySQL.
 - Oversaw release processes from early candidate to operations handoff.
- **Arzoo, Inc.** - Release Engineer - 4/15/1999 - 10/10/2000
 - Learned the role of Release Engineer on-the-fly and actively researched best practices, making informed proposals to development and operations teams.
 - Maintained CVS source tree and existing build/release system.
 - Eventually rewrote build system in Perl with a new system designed by myself with the input of development and operations, which involved tying together disparate build components to be run as a single command across several architectures including Solaris, Red Hat Linux, and Cygwin.

Education

- Ohlone College - Mathematics