

Alex Muller

alex@mullr.net • <http://alex.mullr.net> • <https://github.com/alexmuller>

I love writing software for the web that solves a problem. I use simple, well understood technologies that are appropriate for the task in order to create small applications that can be loosely joined with APIs and progressively enhanced in a web browser to create excellent user experiences.

I'm interested in the creation of video and audio media, especially in how technology is aiding distribution. I am passionate about using the web to enable free and open access to information, particularly through contributing to projects like Wikipedia and OpenStreetMap. I'm increasingly interested in [preserving the history of the web](#).

Work and experience

Government Digital Service (May 2013 onwards)

From May 2013 to December 2014 I worked on the Performance Platform, helping government service managers access and understand the data behind their transactions. I made use of progressive enhancement to display data in as many browsers as possible and helped manage and maintain the Performance Platform infrastructure.

TOOLS: Git, Puppet, JavaScript, Sass, Python, PostgreSQL, MongoDB, Vagrant

ON THE WEB: <https://www.gov.uk/performance> and <https://github.com/alphagov>

In January 2015 I joined GOV.UK's infrastructure team, providing operational support and working on tooling to help improve the stability of the UK government's website while reducing costs.

TOOLS: Git, Puppet, Ruby, Go, VMware vCloud

ON THE WEB: <https://www.gov.uk/> and <https://github.com/gds-operations>

Research and Development software engineer, News International (July 2012 to May 2013)

After graduating, I joined the small R&D team at News International. Using a variety of tools (including Ruby and Node.js) I helped create lightweight prototypes and proof-of-concept apps for *The Sun*, *The Times* and *The Sunday Times*, as well as monitoring and maintaining some apps in production. My favourite project was a marketing campaign titled Feel Good Piñata, a Node.js app with a real-world component that used WebSockets to provide a queuing system for users. I experimented with responsive design as well as designing for different media such as Internet-connected receipt printers.

TOOLS: Git, Ruby on Rails, Haml, Sass, JavaScript, Heroku, Amazon EC2, Node.js

ON THE WEB: <http://web.archive.org/web/201301/http://labs.newsint.co.uk/showcase>

Python and front-end web developer, GlaxoSmithKline R&D (July 2011 to September 2011)

Over the summer between my industrial placement and final year at university I maintained an internal web application at GSK's Medicine Research Centre in Stevenage. The application was originally built predominantly using Python and JavaScript, and I was responsible for fixing bugs and adding new functionality as required.

Web & multimedia communications placement, GlaxoSmithKline (July 2010 to July 2011)

During my placement year I worked on [GlaxoSmithKline's corporate website](#) as a member of their Global Media team. The placement involved writing HTML, CSS and JavaScript (primarily using jQuery) and working with non-technical stakeholders to understand and implement their requests. I was responsible for gathering information in preparation for a project to redesign the site, for example through interviewing users and creating surveys. A reference is available from GSKUK.HR@acs-inc.com.

Education

BSc Computer Science and Mathematics (with a year in industry),
University of York, 2008 to 2012

DEGREE CLASSIFICATION: Second Class Honours, Division One (2:1)

DISSERTATION: [Allocating optional modules to University of York students](#) (79%)

My final-year project involved the creation of an application that allocates optional modules to students after collecting their preferences via a web interface. The creation of this application was supported by the University of York's University Teaching Committee.

The project covered two broad areas of computer science; the construction of the web application (which included gathering requirements, database design, user experience, testing & security), and performing the allocation based on students' preferences and factors like the number of students in a class, which is a constrained optimisation problem.

The application was trialled in March 2012 by the Archaeology & History departments and successfully allocated modules to 800 of their students. The pilot was evaluated as successful and the application was used in the following academic year, supported by the University's IT Services.

A reference for this project is available from [James Cussens](#), Department of Computer Science.

St Paul's School, London, 2003 to 2008

A Level: Computing (A), Mathematics (A), Physics (C)

AS Level: French (A)