Daniel Mason

Personal Statement

Daniel has two software development related degrees and is staying in practice through personal projects and providing additional value to his employer.

Daniel constantly looks at how new technologies and techniques can be used to improve the quality of his work. At Apolitical, Daniel designed a simple deployment mechanism for kubernetes projects. At MOO, he evangelised Docker and CD Pipelines to increase developer confidence and code quality, as well as pushing for more testing and quality control. He's been influential in several cross team guilds. Daniels favourite achievement was mentoring a colleague from a non-engineering team to become a Junior Software Engineer on his team.

Daniel can pick up new languages, frameworks and techniques very quickly, and loves to do so. He constantly looks for new ways to improve on what he's already doing.

Skills

Daniel is a skilled engineer, with experience in a wide range of languages. This is an incomplete list of technologies Daniel has been practicing in the last year:

Languages:	Rust, JavaScript, ES6+, PHP, MySql/Maria, HTML, CSS, SASS
Quality Control:	GitLab CI, CircleCI, Travis, PhpUnit, Behat, PHPMD, PHPCS, ESLint, Mocha, Chai, Sinon
Frameworks:	Express, React, Redux, Aye Aye Api, Symfony
Source Management:	Git, GitHub, GitLab, Cargo, NPM, Composer, Packagist, Docker, Artifcatory
WebOps:	Kubernetes, AWS, DO, GCe, GKE, Docker, Vagrant, StackPoint
Other Tools:	IntelliJ Idea, VSCode, Webpack, Babel, Zookeeper

Experience

Apolitical Group Ltd - Software Engineer Nov 2017 - Today

Daniel designed and engineered the kubernetes infrastructure that is replacing the WordPress Monolith. This is broken up in to different namespaces, each guarded by different levels of security. The public namespace is open to all, it hosts a proxy to the old wordpress site, as well as public microservices such as sign up, authentication, inviting people, and managing profiles. The internal namespace is guarded using mutual authentication certificates. It hosts company wide resources. The engineering namespaces is also guarded by mutual authentication (with a different CA), it hosts an EFK stack and other engineering tools.

Several cloud providers were tested before settling on GKE for both stability and TLS passthrough at the Load Balancers (required for the proxy to pass on the users address).

Daniel's other achievements include; writing open source tools, including a Rust based tool for parsing environment variables into kubernetes opaque secret resources. He also helped the team adopt Node, learn JavaScript best practices, and made sure all PRs get reviewed. He's currently trying Rust for microservice backends.

MOO Print Ltd - Software Engineer May 2016 - Nov 2017

Daniel was hired as for his PHP skills, but ended up working almost entirely on the frontend. He helped take a greenfield project from the drawing board and into production. The project was a React + Redux application using Webpack and Babel, which meant Daniel had to learn a lot about modern JavaScript development very quickly. This turned out to be very successful though and before he left he was teaching others how to develop applications on the frontend, and was an influential member of the Frontend guild, helping build and manage the MOO Design System (MOODs).

Production capability and code quality are both very important to Daniel. To aid with this in his team, Daniel created two pipeline systems for testing code and building artifacts. The first of these was in Jenkins, pipeline plugin, and writing the stages in a JenkinsFile using the Gherkin. To get this working, Daniel built a specialised environment using Docker. The server itself simply ran the Docker Service, then a custom Jenkins image was built with DinD. That service then communicated back to its own host. By using a one to one volume mapping on $JENKINS_HOME, Jenkins could share CI files into containers that would run specific tests.

The Jenkins pipeline was later replaced in favour of GitLab CI. Daniel worked with other crew members to bring the pipeline from simple continuous integration, to a full continuous deployment pipeline. Now code is deployed to a live environment whenever a merge to master occurs. There is still a manual release step but all it does is tell ZooKeeper when the new files are. This means when the decision is made to put new front end code live, it can take as little as 15 seconds from the click of a button in GitLab, and it can be reversed just as quickly.

Daniel's favourite achievement during his time at MOO was not something specifically technical. Daniel was given the opportunity to mentor someone from a non-engineering team and help them move to Junior Software Engineer over the course of two months. During that time they spent one day a week, pairing and creating a small SPA from scratch, covering all of the basics of ES6, React, Redux, Webpack, SASS (and CSS), Babel, HTTP, REST, ESLint, Karma, Mocha, Chai, Sinon, Enzyme, Docker, GitLab CI, Git, and IntelliJ, as well as how to find out anything else they needed to know by themselves. It was an enormous amount for them to get through, but at the end she joined his team and was immediately a significant contributor.

The Foundry Visionmongers - Web Developer Oct 2014 -May 2016

Daniel was seconded to the Made With Mischief team. As the teams sole developer for web services Daniel's role covered almost every aspect of development from server management to PHP to Frontend.

The website was still under construction by a third party when Daniel joined so he initially spent his time writing Behat tests, helping him understand the processes and setting up a quality control system that would continue to be used until Mischief's shut down this year.

Daniel built a new API using Aye Aye Api to quickly produce a fully RESTful interface that provides functionality to the website, the desktop app, and another unreleased application. App developers liked how easy the Api was to use due to the self documenting feature of Aye Aye.

Loft Digital - LAMP Developer Sept 2011 – Oct 2014

Work primarily involved developing server side functionality using PHP and MySQL, but also included working with JavaScript to provide client side functionality as well as creating and maintaining servers.

During a project to refurbish the "find a store" feature on Cyclescheme's public website, used a framework he had been building at home. This was a precursor to Aye Aye Api and was so successful, it was used on Business Traveller, Cycle To Work Day, Craigmore, the internal Accounts system and the Homeless World Cup. While the core ideas have stayed the same Aye Aye was rewritten almost entirely for scratch and is now available under the MIT license.

Daniels biggest achievement was working onsite at the Homeless World Cup in Poznan. Daniel provided extra value to both the sports and media teams. The role involved developing new features, as well as diagnosing and resolving problems that developed during the event.

Contact Details

Mobile:	07838 200176	Twitter:	@Gisleburt
Email:	daniel@danielmason.com	GitHub:	https://gisleburt.github.io/