Seb Insua *Engineering Lead & Consultant Engineer*

me@sebinsua.com • 07912303758 • github.com/sebinsua • sebinsua.com

Summary

Consultant Engineer.

I learnt to program at the age of sixteen after discovering a flourishing homebrew community. Realising that self-taught amateurs could create products was inspiring to me. I'm still amazed by what small empowered teams achieve.

I am comfortable in almost any kind of software engineering position, whether it's developing native apps, tooling, data pipelines, back-end systems or complex real-time applications. I do a lot of commercial work in JavaScript, but in my spare time love developing with Rust due to its unique mix of modern syntax and low-level programming.

I used to do a lot of functional programming, but nowadays I just try to borrow the concepts and write the most beginner-friendly imperative code I can. Overall, my engineering practices are influenced by the twelve-factor app, a conservative attitude towards microservices, and Simple Made Easy. I follow the mantra *the right tool for the right job* and stay as simple and common as possible.

I am deeply focused on *finding the right problem*.

Knowing when not to solve a problem is as important as knowing how to solve it.

Technical specialties: Full-stack software architecture and implementation.

Experience

JPMorgan Chase & Co.

Software Engineer / Architect *Aug '19 – Oct '20*

- Advised on the technical direction of the group and provided solutions to difficult organisational problems.
- Created caching infrastructure to speed up CI. In one case, builds that previously took 19 minutes took 3 minutes.
- Helped with the upgrade of a large 50+ component library from v3 of Material UI to v4.
  - Introduced TypeScript.
  - Introduced visual regression testing.
    1. Auto-generated 100s of tests and then reduced the time these tests took to run from 25 minutes to 10 minutes by developing a codemod that automated rewriting 10,000s of lines of application code to use code-splitting techniques.
    2. Contributed to the open-source libraries 'cypress' and 'cypress-image-snapshot' and wrote cross-platform code and dockerized Cypress so that it could be used by engineers on Mac, Windows and Linux machines (to avoid image regression failures due due to OS, browser, and font rendering differences).
    3. Used the container-native workflow engine Argo to parallelise the tests on a Kubernetes cluster. This effectively allowed test runs that would otherwise take hours, to run in under a minute.

Shell — London, United Kingdom

Technical Lead *Dec '18 – Jun '19*

Shell is one of the six oil and gas "supermajors" and the fifth-largest company in the world.

- Laid the foundations of a scalable Data Science/Engineering practice.
- Starting from zero, hired multiple Data Engineers and Data Scientists into the team (approximately 10 DE/DS). Established a streamlined hiring process for the future.
- Reviewed earlier work on the project by a consultancy, and created a plan on how to improve problem areas. Communicated the path forwards to both non-technical stakeholders and globally distributed technical team members. Centred power within a core DE/DS team that would oversee future quality, and setup weekly discussions for thoughtful collaboration and daily stand-ups to stay in sync.
- Established a continuous integration process to allow robust improvement of the data engineering and models. This was used to test and document the original code so that it would be understood by future team members.
- Prioritised the rapid build of a scalable data platform to enable the new development workflow. This caused the data pipeline to run end-to-end 4x faster, and meant that data engineers could spend more time programming and less time waiting for pipelines to finish.
- Helped the team to share their knowledge of modern software engineering practices. Together we created and taught training workshops to teach Data Scientists modern practices for collaborating towards high-quality models (Git, PRs, CI, etc). Outside of the workshops, we used shared repositories to document the onboarding process, how to use Git for collaboration, best practices in EDAs, etc.
- Tools: Azure, CircleCI, Docker, Helm, Jupyter, Kubernetes, Python, Spark Cluster, Written Word.

JPMorgan Chase & Co. — London, United Kingdom

Application Engineer *Jun '17 – Dec '18*

JPMorgan Chase & Co. is an American multinational banking and financial services holding company.

- Helped to lay the foundations necessary for the high-scale software development of a complex realtime trading application built using TypeScript and React.
- Improved observability of software development by introducing a CI/CD environment that supports rapid feedback on real work by automatically deploying sites and posting comments into PRs linking to these.

- Improved modularity of software by demonstrating how to use Rollup and Babel 7 to create best-in-class packages that support CommonJS, ES modules and types.
- Improved encapsulation of development work by making an authenticated and declarative API for the loading and code-splitting of sub-applications.
- Used working code to teach best practices on higher-order components, render props, reducer components, performance implications, hooks, etc.
- Evangelised a modern development process leading to multiple teams across the company moving to Lerna monorepos.

YLD — London, United Kingdom

Node.js Engineer — *Jan '17 – Apr '17*

YLD is one of London's fastest growing software engineering consultancies.

- Code reviews. Troubleshooting framework issues. Implementation of 4 Node.js microservices.
- Created shared helpers to ensure a set of data exists during test execution.
- Improved the error handling and logging of an in-house microservices framework.
- Wrote a bit of documentation explaining the process of migration between SQLite and MySQL.

McKinsey & Company — Western Europe

Full-stack Engineer — *Apr '16 – Dec '16*

McKinsey & Company is a worldwide management consulting firm.

- Code reviews. Mentoring. Documentation. Architecture.
- D3. React. Redux. Node.js.
- Wrote a number of open-source projects (e.g. redux-saga-helpers and react-redux-wizard).

The Economist — London, United Kingdom

Full-stack Engineer — *Sep '15 – Dec '15*

The Economist is an English-language weekly newspaper.

- React. ES6. Node.js.
- Numerous components (e.g. @economist/component-articletemplate) used in The World In 2016 project and potentially other projects.

Home Office — London, United Kingdom

Technology Lead — *Jan '15 – Sep '15*

The Home Office is a ministerial department of the Government of the United Kingdom, responsible for immigration, security, and law and order.

- Rebuilt a sprawling legacy system into 6 Node.js microservices harnessing infrastructure, services and libraries built for the Passport Exemplar project.
- 1 JWT Authentication API with LDAP as its backing service; 1 old-school Express.js front-end app; 1 single-page app built with React + Flux + ES6 (Babel); 2 RESTful Express.js APIs; and 1 command-line app to run tasks on the Postgres database.
- Gave strategic advice on the architecture of a new project, and later presented this plan at 'Show and Tell' to a wide range of people throughout the organisation.
- Created some early prototypes using Java 8 + Dropwizard + Swagger.

Red Badger — London, United Kingdom

Node.JS Consultant — *Mar '14 – Jul '14*

Red Badger is a creative software workshop.

- This was a client-facing role in which I provided training to an external team.
- Node.JS, RabbitMQ, Elasticsearch, Redis.

Hailo — London, United Kingdom

JavaScript Engineer — *Sep '13 – Jan '14*

Hailo is the evolution of the hail – a free smartphone app which puts people just two taps away from a licensed taxi, and lets cabbies get more passengers when they want them.

- Created a series of single-page dashboard apps. Conversations with internal customers helped me to come up with ideas on how to simplify business processes. For example a way of signing off groups of driver's profile changes was given a UX that defaulted to approval but forced the user to sign a receipt of the changes: this helps with speed while also ensuring accountability.
- Other components were created so that behaviour relating to presentation was well-separated from configuration and data input; or made so that they could be updated in real-time, for example: an animated map of driver locations.

BIZZBY — London, United Kingdom

Senior Node.JS Engineer — *Apr '13 – Jun '13*

Bizzby is app your service to help you book a trusted local service in 30 seconds.

We R Interactive — London, United Kingdom

Team Lead — *Oct '12 – Apr '13*

We R Interactive blends the best of games, film and TV production to create social games that bring global audiences together around sport and music.

- Designed and led the creation of a back-end system to support a sport game as well as recruiting developers for my team.
- The system was decomposed into multiple services and stored data in Cassandra. I created a service which would automatically generate a series of questions from a stream of data and a template, a real-time market-outcome resolution service that would automatically resolve a series of previously generated questions depending on a simple DSL and a stream of real-life data, and a service that could have data pushed to or pulled from it and parse this data into a common format before feeding it into a message queue for deeper processing.
- Setup a continuous integration system using Jenkins, and used Puppet to automate deployment of some of the services to AWS.

Saffron Digital — London, United Kingdom

Team Lead — *Oct '10 – Oct '12*

Saffron Digital is the global, market-leading provider of connected device video, DRM, advertising and platform services. Saffron Digital was acquired by HTC in 2011.

- Architect on the HTC Watch project in which I later led a team of five developers. Due to my early work on internationalisation and implementation of multiple payment services we were able to scale this up to 20 countries.
- Significant input in re-engineering development process as we moved from being a startup to a larger company. For example: an engineering culture; Git instead of SVN; continuous integration; modern deployment tools; etc.
- Architecture and development of a new platform based on understandings gleaned from previous services. Worked on service to orchestrate and configure encoders in order to run encodes in parallel on AWS.
- A client-side application for Samsung Connected TVs and set-top boxes written in object-orientated JavaScript.

PHP Developer — *Feb '10 – Oct '10*

- A RESTful web service for FOX to support a fully localised Family Guy video streaming Android application. A CMS was also created as well as reporting tools to give the business metrics to measure activity.
- A web service for Paramount Studio s the League of Extraordinary Dancers iPhone app. This was interoperable with the iTunes video store in order to check receipts.

Please refer to my Linkedin profile for the complete list of work experiences along with recommendations.

Education

Coursera — Online

Machine Learning — *Feb '16*

Course Certificate, License E3XLGER56CQ3

University of Kent — Canterbury, United Kingdom

Bachelor's degree in Computer Science — *2005 – 2009*

Cranbrook School — Cranbrook, United Kingdom

GCSEs & A-Levels (secondary education) — *2000 – 2005*

Interests

Business, Economics, Poetry.