

EXPERIENCE

Co-creator & Software Engineer, Apex (2020)

Remote (based in London, UK)

Open-source API Proxy for microservices that provides one place to log and control all service-to-service traffic

- **Researched problems:** identified challenges of working with microservices by reading books on microservices and company engineering blog posts, and watching seminar presentations
- **Compared existing solutions:** researched the trade-offs of API gateways, microgateways, service meshes, container orchestrators, workflow orchestrators, distributed tracing, cloud config servers and client libraries
- **Designed architecture:** decided on simple architecture of proxy server + configuration store + logs database, targeting the needs of small teams migrating from a monolith to their first few microservices
- **Built proxy server:** chose Node.js for its ability to handle high number of simultaneous connections with high throughput, its widespread usage, and the extensibility of the Express.js framework, over NGINX and Envoy
- **Extended proxy with middleware:** built middleware layers for authentication, routing, logging and retry logic
- **Connected time-series database:** chose TimescaleDB for its high data ingestion rates, and configured it store logs of all requests and responses passing through proxy
- **Connected key-value store:** chose Redis for its high read rate, and configured it to store proxy's config data
- **Built admin API:** built Node.js and Express.js API for proxy, logs database and config store
- **Built admin front end UI:** built React UI to communicate with admin API
- **Deployed components in containers:** containerized components with Docker, and co-ordinated their deployment with Docker Compose
- **Deployed containers to AWS:** configured containers for deploying to AWS Elastic Container Service
- **Co-authored technical case study:** explained the problems solved by Apex, compared existing solutions, and described Apex's design and implementation at <https://apex-api-proxy.github.io/>

Full-stack Web Developer, Self-Employed (2018 - 2020)

Remote

Open-source web applications using Ruby, Node.js, PostgreSQL, HTML, CSS, JavaScript

- **Rello:** a Trello-like Kanban board with back-end API built with Rails, and front-end UI built with React + Redux
- **Post It:** a Reddit-like social news and discussion app built with Rails
- **Ticket Tracker:** a project- and ticket-tracking app built with Rails
- **Airline Routes:** a filterable database of airline routes built with React
- **Todos:** a to-dos tracker built with Node.js, Express.js, JavaScript, jQuery, Handlebars, HTML and CSS
- **Book Viewer:** an e-book viewer built with Ruby, Sinatra, ERB, JavaScript, HTML and CSS

Co-founder, Parent Genie (2017 - 2018) & Humanry (2016 - 2017)

Hong Kong

Q&A platform for parenting advice; B2B marketplace for executive coaches; Mobile micro-learning platform

- **Design sprints:** ran design sprints to rapidly design, build and test core features with actual users
- **Built mobile app MVPs:** built and shipped apps to app stores in 2 weeks; tested products with 40+ users

EDUCATION

Launch School - Software Engineering School (2018 - 2020)

Remote

Multi-year full-stack Software Engineering curriculum focused on mastering web development fundamentals

- **Back end:** Ruby, Node.js, HTTP, PostgreSQL; **Front end:** HTML, CSS, JavaScript

Trinity College, University of Cambridge (2011 - 2015)

Cambridge, UK

BA Mathematics (3 years) & Management (1 year, at Cambridge Judge Business School)

- **Class I** in Mathematics (years 1 and 2); **Class II.1** in Mathematics (year 3) and Management (year 4)

TECHNOLOGIES

Back end: Ruby, JavaScript (ES5, ES6), Node.js, SQL, PostgreSQL, Redis, TimescaleDB, Docker, Express.js, Sinatra, HTTP, REST APIs, AWS ECS & Fargate, Git; **Front end:** Rails, React, Redux, jQuery, HTML, CSS