

# Andy Kuszyk CV

## About Me

I am:

- An **experienced engineer**; with a decade of individual contributions in Golang, Terraform, AWS, Python and .NET.
- A **competent leader**; I have grown teams and projects from inception; I have led individuals, and entire engineering organisations.
- A **confident architect**; I have designed new systems and platforms from scratch, and have facilitated a culture of excellence in growing teams.

I value:

- **DevOps** practices and philosophies.
- Strong **writing cultures** for documentation and design.
- **Emacs, Linux**, and free software!

You can read more about me on my technology blog: [andykuszyk.github.io](https://andykuszyk.github.io), or contact me on LinkedIn: [linkedin.com/in/andy-kuszyk](https://linkedin.com/in/andy-kuszyk)

## Primary Skills

- Application development in **Golang**, and **Python**.
- Infrastructure development in **Terraform**.
- Platform design with **AWS** and **Kubernetes**.
- Telemetry management with the **Prometheus** and ELK stacks.

---

## Work Experience

### Deliveroo (2023 - present)

#### Staff Platform Engineer (Nov '23 - present)

I have recently joined Deliveroo as a staff engineer in their networking, edge, and compute team.

### Form3 (2019 - 2023)

#### Staff Engineer (Nov '22 - Nov '23)

At the end of 2022, I decided to take a step away from engineering management, back into an IC role. I spent a year as a Staff Engineer on a project building payment systems for the US market.

In this staff role, I am was mostly self-led; I sought out any issues in the team's practices or codebases, and try to help improve them where I can. I often paired

up with members of the team to help solve specific problems, or to plan larger programmes of work.

After more than four years at Form3, I decided to spend some time freelancing to gain more experience of different companies at different stages in their growth, and engineering organisations at different levels of maturity.

**Notable projects:**

- Wrote a GitHub Action to synchronise classic and v2 project boards.
  - Team project management was scattered across multiple classic and v2 GitHub project boards.
  - I consulted with the teams involved, and wrote a GitHub action using Bash, Python, and the GitHub GraphQL API to automate status updates to a single v2 board.
- Facilitated the architecture review and programme planning for a major API resource change.
  - Product teams had determined changes were required to the API surface for a new functional area.
  - I assembled a working group of engineers, and led the architecture review of the changes.
  - In collaboration with the working group, I planned a programme of work to implement the changes, and ensure that they were sufficiently well documented to be adopted by other internal teams.
- Database schema migration library changes.
  - The library used for database schema migrations was central to the way our applications deployed schema changes, but it had various problems during the lifecycle of our programs, and didn't scale well.
  - I performed a deep-dive into the library's internals, reviewed alternatives, and contributed a new version of our framework that mitigated the problems we had observed.
  - I rolled out the changes in my functional area, and paired with colleagues in other teams to upgrade their applications.
- Consolidated team and project documentation from many disparate sources into a single, indexed location.
  - When I joined the project, access to documentation was clearly a problem for team members.
  - Documentation was spread amongst wikis and markdown files in several repos, as well as in documents on Sharepoint sites.
  - After consultation with the team, I moved all the documentation into a single location, stored as markdown files in a Git repo.
  - I wrote and implemented a tool for indexing documents based on tags, in order to generate dynamic READMEs for different folders (see [github.com/andykuszyk/markasten](https://github.com/andykuszyk/markasten)).

## **Head of International Engineering (Feb '21 - Nov '22)**

In 2021, I started working on a new project at Form3 to build out a fresh product line for international payments.

This involved building applications in Golang, AWS infrastructure with Terraform, and a compute platform in Kubernetes.

It also gave me the opportunity to shape how an engineering team could work and grow, and allowed me to introduce good foundations for knowledge sharing, documentation, and asynchronous decision-making.

In this role I:

- Laid the groundwork for three engineering teams to build out a product successfully with high-quality CI pipelines, load testing, observability, and functional testing.
- Provided technical leadership, and project management.
- Led and facilitated design work for future development.
- Advocated and supported asynchronous knowledge sharing, documentation, and decision-making (using PRDs, RFCs, and ADRs).

### **Notable projects:**

- Conducted extensive research into a new product area, and produced initial system, platform, and application design blueprints.
  - This involved reviewing a large amount of third-party documentation, and summarising it in internal documentation.
  - Various design options were possible, and were explored via iterations on RFCs and ADRs.
- Built a platform proof of concept for a private, AWS/EKS Kubernetes cluster.
  - In order to meet strict security requirements for a third-party, we needed a Kubernetes cluster with heavily restricted internet access.
  - I built a proof-of-concept in Terraform for a cluster, VPC, routing table, and associated networking resources that facilitated restricted access to the control plane via a bastion, and filtered network egress via a Squid proxy.
- Grew an engineering team from 1 to ~25 engineers.
  - Established initial good practices for onboarding, knowledge documentation, and asynchronous decision-making, to help the team grow rapidly without reducing build velocity.
  - Paired extensively with new joiners, and then later coached a team of lead engineers, to maintain high standards, and a culture of self-organisation.
- Led the architecture and design of a distributed software system operating across multiple Kubernetes clusters.
  - Initially led, and then later facilitated, design of new functional areas with working groups of engineers.

- Software systems interacted with Postgres databases, exchanged messages via SQS, SNS, and S3, and made use of a EFS volumes exposed to pods as Kubernetes volumes.
- Managed the project delivery of a large engineering effort over a two year period.
  - Planned, and estimated, future milestones, keeping track of project progress on a detailed project plan.
  - Collaborated with product team members, and lead engineers, in order to maintain alignment of project delivery with the plan.
  - Regularly briefed the engineering teams, and shared project delivery widely, so that individuals were empowered to make educated choices about next priorities, and where to spend effort.

### **Senior Engineer (June '19 - Feb '21)**

When I joined Form3, my first role was a member of the scaling and performance team. I spent most of my time working on Golang and Java code bases, and on analysing the performance of a distributed payments processing system.

This involved lots of work with instrumentation (generally Prometheus), as well as working on our in-house load testing tool: [github.com/form3tech-oss/fl](https://github.com/form3tech-oss/fl).

The workloads I was analysing ran on ECS or Kubernetes, and most relied on asynchronous message processing via SQS or NATS.

#### **Notable projects:**

- Scaled platform performance from <10 transactions per second (TPS) to 680 TPS.
  - Ran a series of high-volume tests over the course of several months to scale platform horizontally in order to meet customer requirements of 680 TPS.
  - Used the K6 load testing tool, along with custom internal tools to interface load tests with SQS and SNS.
  - Routinely resorted to hypothesis-based debugging and troubleshooting to identify performance bottlenecks, and contribute fixes.
  - Wrote and presented detailed reports on system performance for external stakeholders.
- Introduced distributed tracing to system.
  - Tracing was already present, but trace contexts were not being effectively propagated across network and language (Java/Go) boundaries.
  - I identified problematic propagation boundaries, and contributed changes which enabled traces to extend throughout the entire distributed system.
  - As a result, tracing became a valuable debugging tool for understanding overall system performance, rather than being limited to individual programs.
- Introduced continuous, synthetic traffic to pre-production environments.

- In order to catch performance-related problems before they appeared in production, I introduced continuous traffic in pre-production environments.
- I reviewed real traffic patterns from our production environments, and modelled them with a Gaussian function in a Python script (developed using a Jupyter Notebook).
- When I was able to approximate real-world traffic, I used the results of my analysis to apply traffic to our environments at volumes that were forecasted for six months in the future.
- Was a core contributor in the development of an in-house load testing tool ([github.com/form3tech-oss/fl](https://github.com/form3tech-oss/fl)).
  - K6 had various limitations which meant that it did not satisfy all our needs.
  - In the end, we decided to write our own load testing tool.
  - I started the project, and was a core contributor to both the core load-testing engine, as well as a number of the peripheral load-testing modes.

## **CarFinance247 (2016 - 2019)**

I joined CarFinance247 as a DevOps engineer to help the team transition from a single monorepo with no automation, to a modern Git-based estate with automated deployments.

Throughout my time at the company, I became a Lead Engineer of a team building systems that interfaced with third parties, and then Head of Engineering.

In the role of Head of Engineering, I led five engineering teams, and a data science team. I was heavily involved with the start of a new lender–247Money–and worked closely with the data science team to build effective credit scoring machine learning models, using a DevOps approach.

## **Galleria (2014 - 2016)**

I joined Galleria as a Junior Engineer, and worked on .NET desktop applications. I spent most of my time at the company working on a back-end space optimisation algorithm.

## **Capita (2012 - 2014)**

My role at Capita was an MI Analyst, which mainly involved analysing data in Excel. As a keen amateur programmer, I used VBA extensively to automate Excel operations, and wrote a number of small VB.NET and C# front-ends to databases.

## **Teaching (2010 - 2011)**

Prior to my technology career, I trained and worked as a secondary school Physics teacher.

---

## **Education**

- MSc in Advanced Computer Science (MMU, 2016 - 2018)
- QTS in Secondary Education (University of Reading, 2010 - 2011)
- BSc in Physics (University of Reading, 2005 - 2008)