



Daniël Koek

contact@danielkoek.net 0830981800

Residence Bandon, Co. Cork,
Ireland

Nationality Dutch

Birthday 21-12-1994

Driver licence Full Irish

SKILLS (IN YEARS)

C# .NET/ ASP.NET 9

Front-End Technologies 10

Azure Devops 8

.Net Core 8

Docker 8

Kubernetes 8

SKILL SET

As a full stack developer, I have extensive experience in both back-end and front-end technologies. This allows me to adapt and balance these fields effectively. Additionally, I have significant infrastructure (as code) expertise, enabling me to transform concepts into fully live applications.

Knowledge Front-End

- LIT
- Cloudflare workers (serverless)
- TailwindCSS
- NPM/NodeJs - both use and maintain
- JavaScript - any ECMA version
- Web workers
- Web-Components
- React/Remix
- Angular (incl. SSR)
- VueJS

Knowledge Back-End

- C#/.NET - including .NET Core
- Entity Framework
- Elasticsearch
- Various message brokers (Kafka, RabbitMQ, Azure Service Bus)
- Rest API, Middleware
- Microservices and Monoliths
- Git
- Kubernetes (incl. custom controllers/CRDs)

Knowledge DevOps

- ELK/Logging systems
- PostgreSQL, MySQL
- Azure DevOps, GitHub Actions
- Azure (AKS, Service Bus, Front Door, Key Vault)
- Infrastructure as Code (Bicep, Helm)
- Docker
- OpenTelemetry, Prometheus, Grafana
- Serverless architectures

Work Experience

Introduction

I am a full stack developer with C# at the core and close to a decade of experience across the three layers that make software actually ship: front end, back end and the infrastructure underneath. I have deliberately worked in many different industries and company structures, which taught me the ups and downs of most technology stacks — and how to pick the right one for the job, rather than the fashionable one. I am highly adaptable and quick to find my way in an existing codebase. The things that I strive for:

- **FrontEnd:** HTML/CSS/JS started as a means to get a button to do something in the back end, but I quickly learned that good UI is what makes data useful — and often harder to get right than the backend. A CRUD is easy; the fun challenge is presenting complex concepts so a user understands them without needing to understand the system. I have in-depth vanilla JavaScript experience (including fun side projects built without any framework), which makes any framework easy to pick up
- **BackEnd:** Deep understanding of C#, keeping up with the latest language versions. I write code that the next person immediately understands, using a minimal set of essential, well-maintained NuGet packages. I understand most other languages — C/C++, VB, Java, Python — just enough to convert them to C#!
- **Infrastructure & DevOps:** Kubernetes, Docker, Helm and infrastructure as code, with strong CI/CD pipelines that keep deployments boring, reliable and the development cycle short
- **Solution architecture:** Hands-on experience with monoliths, microservices and serverless — and, more importantly, knowing which one the business actually needs. Always with security and observability in mind
- **Maintainability:** Readable code that stays optimised for speed and reliability. I quickly understand existing code and add features that fit the narrative of what is already there, backed by Domain Driven Design and solid testing principles
- **Delivery:** Iterative, feedback-driven delivery — small previews early, unit tested from the start, shipped often

Uniphar (Private LTD, Greenfields-Innovation) (Remote)

Period	February 2025 – Present
Branch	Warehouse management
Title	Full stack developer
Methods & Techniques	Scrum (AzureDevops)
Systems Used	Azure DevOps, AKS, Docker, Azure Service Bus, Azure Front Door, Microsoft Teams, ServiceNow, Azure Data Explorer
Languages	C#, bicep
Frameworks	Kubernetes, OpenTelemetry, Entity Framework

At Uniphar I work in the Platform Engineering team. Together we build and maintain the shared services that every other product team depends on to move data and files reliably through the organisation. Rather than owning a single product, we look after a whole set of platform components — a Kubernetes ingress controller, a message broker monitoring platform, a Teams notification API, and FrontGate, the company's file onboarding and processing solution — and I move between them wherever I am needed most. Across all of these the focus is the same: reliable, observable services on modern .NET (currently .NET 10) running on AKS, with a strong emphasis on OpenTelemetry so issues surface before they become incidents.

The main tasks I had at Uniphar were:

- Contributing to FrontGate, the company's onboarding and file processing solution, adding new ingestion connectors, fixing production issues and migrating its Kubernetes ingress handling onto a custom AbstractIngress CRD
- Building and maintaining the message broker monitoring platform, which watches Azure Service Bus queues and Logic Apps/Integration Suite for failures and surfaces them proactively in Teams
- Developing the Teams Notifications API, which provisions Teams apps and posts interactive cards (reprocess/view-file actions) back into channels
- Working on IngressProcessor, the AKS controller that injects Kubernetes ingress configuration into Azure Front Door, including designing the AbstractIngress CRD now shared across other platform services
- Building a ServiceNow integration API for creating and managing incidents, and a SAP event ingestion service that sinks events into Azure Data Explorer
- Contributing to shared platform building blocks: the Uniphar.Platform.Telemetry NuGet package, the Document API, DotFlyer (SMS/email sending) and the pod notes controller used for AKS image cleanup
- Rolling out OpenTelemetry, health probes and shared CI/CD workflow conventions across these services, and tightening workflow permissions as part of a broader security hardening effort

Kiwa (Freelance) (Remote)

Period	December 2023 – January 2025
Branch	Certification of meat(pigs)
Title	Full stack developer
Methods & Techniques	Scrum (AzureDevops)
Systems Used	Azure DevOps, Entity framework, .net framework, .net core,
Languages	C#, jQuery
Frameworks	EF & Kendo

Kiwa asked me back — not for the project I was originally hired for, but to give some of their internal systems the TLC they needed. Alongside day-to-day maintenance and new features, I drove a migration from .NET Framework to .NET Core: with minimal effort I removed the dependencies that were blocking the switch and successfully migrated the application. Because I learn a new system inside out very quickly, I also became the go-to person for a number of older applications nobody in the company had knowledge of anymore, and I wrote the documentation that was missing.

The main tasks I had at Kiwa were:

- Maintaining and extending the internal line-of-business applications
- Fixing bugs, building new features and finding bugs before users did
- Cleaning up legacy code, upgrading projects and designing upgrade paths (.NET Framework to .NET Core)
- Helping build an invoice proposal system
- Reverse-engineering undocumented systems and making them maintainable again

Vicrea (Contract) (Remote)

Period	September 2022 – December 2023
Branch	Geometry management for local governments
Title	Solution architect/Full stack developer
Methods & Techniques	Scrum (AzureDevops)
Systems Used	Azure DevOps, K8s, Docker, Cloudflare, PostgreSQL
Languages	C#, TypeScript
Frameworks	EF, Remix, React, Kendo, GRPC, serverless, IdentityServer4, GeoServer

When I joined Vicrea, the application was a proof of concept running on local Portainer instances, with little consideration for reliability, security or isolation of business logic. My main task was to get a production-ready application launched. Leveraging my years of experience across software stacks, I turned it into a cloud-native application with the observability you would expect of any modern system. Along the way we removed every manual step — new customer environments are provisioned fully automatically — and made sure the application could handle large amounts of geo-data without incurring a large cost to the company.

The main tasks I had at Vicrea were:

- Creating CI/CD pipelines to automatically deploy new versions of microservices (leveraging helm)
- Isolating the FE from the BE and deploying this using Cloudflare workers in order to have a serverless global low latency way to serve HTML/CSS/JS
- Building logic so that part of the FE was only available for authenticated users
- Automatically using infrastructure as code to provision new environments, including databases, docker registries and k8s clusters
- Refactoring major parts of the existing code to be compliant with Domain Driven Design concepts
- Creating background processes that would pull large datasets from various government agencies. This was then redistributed to various internal APIs, which in turn would provide insights for users
- Removing FE-microservices and creating a mono FE, refactored from single-spa to Remix, introducing Tailwind and moving from Leaflet to OpenLayers
- Mentoring/teaching other developers in kubernetes and how to develop decoupled software

Kiwa (through TeamRockstar) (Apeldoorn, The Netherlands)

Period	June 2020 – September 2022
Branch	Manure management
Title	Full stack .net core developer
Methods & Techniques	Scrum/Kanban (AzureDevops)
Systems Used	Azure DevOps, K8s, Docker, Cloudflare, Kafka, ElasticSearch
Languages	C#, TailwindCSS, JavaScript, Yaml, Soap
Frameworks	AutoMapper, Entity framework, Microservices, pub-sub, Kafka, Tailwind, Web-Components, Lit, Snowpack, Cloudflare, ES, HTTPS, Agro Tunnel, OAuth 2.0

Kiwa tasked us with building a proxy-type application to collect and process data for the manure industry. The system had to be asynchronous, reliable, secure and highly flexible, so we chose containers and Kubernetes — and since I had previous experience there, I designed and set up the foundation of the system. We picked Kafka as the pub-sub backbone for its reliability and Elasticsearch for logging, and built an API-driven front end using utility CSS and Web Components to keep that side just as flexible and scalable. We even built our own OAuth server due to the constraints presented to us. It was the first time I built everything from scratch, and the project proved that 'new' technologies are not just worth it — they are highly flexible and maintainable when done right.

The main tasks I had at Kiwa were:

- Designing, building and implementing the infrastructure from scratch
- Introducing container technologies to a company that still ran servers on bare metal and had no prior history with Docker
- Building backend services (with full CI/CD) to process SOAP, XML and SFTP, exposed through RESTful APIs for a very flexible way to insert and collect data
- Creating a dynamic, data-driven front end — table-heavy screens that had to stay readable, responsive and flexible for future changes
- Creating a full OAuth flow with multiple login systems and roles maintainable from the front end, including a "Permission Module" uniquely developed for the agriculture market but customisable for future portals
- Improving security end to end: Cloudflare Tunnels, E2E HTTPS traffic and a zero-trust security plan with higher standards than any company I had encountered before

Univé (through TeamRockstar) (Zwolle, The Netherlands)

Period	August 2019 – May 2020
Branch	Insurances
Title	Front end Web Developer
Methods & Techniques	Scrum (Jira)
Systems Used	Azure DevOps, K8s, Docker, Nginx
Languages	HTML, JavaScript, TypeScript, CSS
Frameworks	Tailwind, UtilityCSS, LitElement, LitHTML, Snowpack, NPM, jQuery, SASS, Web-Components, Storybook, WebPack, Karma, Visual Testing

When I started at Univé, a lot of older techniques were still in use — jQuery among them — which, combined with hard-to-read code, slowed every developer down. I introduced Web Components (LitElement) together with Tailwind's utility-based CSS, which made the team quicker and more productive without changing too much of the existing workflow, while dramatically shrinking both the JavaScript and CSS on new components. To make it all easier to manage I also brought in Karma for unit testing and Storybook as a component library, so UX could review components at an earlier stage and we all had a better view of what lived where. Together this let us build a more powerful user experience while keeping a consistent look and feel.

The main tasks I had at Univé were:

- Maintaining unive.nl and its logged-in environment
- Introducing new technologies tailored to the product and bringing them to production (Tailwind, LitElement)
- Maintaining and improving build and release pipelines for quicker builds and better stability
- Building new features that still gave a rich user experience despite limited availability of surrounding systems — finding creative solutions rather than being limited by systems outside our control
- Being a helping hand for other team members when they got stuck building new or improving existing code

RTL (through TeamRockstar) (Hilversum, The Netherlands)

Period	February 2019 – June 2019
Branch	Media
Title	Team lead/Full stack .Net core developer
Methods & Techniques	Scrum (Jira)
Systems Used	Azure DevOps, K8s, Docker
DB/DC	Nifi
Languages/Tools	.net core 2.2/K8s/Docker/MassTransit/RabbitMQ

RTL has loads of systems that need to communicate with each other. We built a data hub that funnels a few source systems into multiple target systems, reducing load and making it possible to integrate legacy systems with the newer platforms RTL acquired. The data ultimately serves as metadata for platforms like RTL XL and Videoland.

The main tasks I had at RTL were:

- Receiving and adapting data from the source systems
- Creating distributors that fan the data out to four target systems, each tailored to that system's needs
- Building the CI/CD pipeline in Azure DevOps and setting up Azure resources
- Guiding the team as team lead: introducing Kanban and gathering business requirements from the product owners

MyDataFactory (through TeamRockstar) (Meppel, The Netherlands)

Period	October 2018 – February 2019
Branch	Matching Data
Title	Full stack C#/Angular developer
Methods & Techniques	Scrum (Jira)
Systems Used	Azure DevOps, K8s, Docker
DB/DC	ElasticSearch & MySql
Languages/Tools	.net Core 2.2/C#/TypeScript/html5/Angular/npm

MyDataFactory is a company that specialises in building matching engines for customers that have invoices coming in from different sources and that want to match this to their own in-house catalogue of products. Through the process of manual and automated tagging of potential products, the user can decide what products with what items on the bill, after which the system will improve itself to remove further human intervention.

The main tasks I had at MyDataFactory were:

- Building the front end where matches are suggested, approved and — after processing — downloaded for other systems
- Building and deploying a proof of concept moving the entire infrastructure to Kubernetes as infrastructure-as-code, automated with Helm
- Consolidating the existing tag structure into a single system with deep-dive tests to guarantee tag behaviour now and in the future
- Migrating Azure Functions to Kubernetes pods consuming from Kafka, which improved performance significantly, with results stored in Elasticsearch

PFM-Intelligence (through TeamRockstar) (Alphen aan den rijn, The Netherlands)

Period	June 2018 – September 2018
Branch	Footfall
Title	Full stack C#, Angular Engineer
Methods & Techniques	Scrum (Jira)
Systems Used	AWS Serverless lambda, S3
DB/DC	PostgreSQL
Languages/Tools	.net Core 2.1, C#, Typescript, html5, Angular, npm, yarn

PFM creates reporting, data repair and data collection software to measure footfall (visitors). The old stack had to be phased out, so together with the development team I rebuilt its functionality in .NET Core and Angular — with many improvements and optimisations, partly by improving the data flow and partly by cutting cost through serverless approaches like AWS Lambda.

The main tasks I had at PFM were:

- Building new front end functionality and refactoring existing Angular code
- Creating Lambdas that consume Kinesis events from field sensors, parsing, processing and validating the data before storage

TransUnion, TrustEv (Cork, Ireland)

Period	March 2017 - June 2018
Branch	Anti-Fraud
Title	Full stack C# Engineer
Methods & Techniques	SaFe/Kanban (Scrum)
Operating systems	Windows 10/Cent OS(servers)
DB/DC	ElasticSearch, Hive, HDFS, Logstash, Curator, Hadoop, Kafka
Languages/Tools	Microsoft Orleans, Web Api, C#, JavaScript, html5, Java, Python

We built a fact/score-based fraud prevention system that determines whether a transaction sent by a customer belongs to a legitimate person: let the good guys in, make it very hard for the bad guys, using aggregation over big data sets to identify users based on their history.

The main tasks I had at TransUnion were:

- Integrating third-party data sources (XML, SOAP, JSON and custom formats) for better fact results
- Extending the existing APIs and surfacing them in a VueJS demo application
- Creating test and mock systems for better QA automation
- Improving delivery pipelines to speed up releases and remove manual steps
- Building custom NiFi processors for handling data streams
- Taking part in the on-call rota for outages, remotely diagnosing why systems went down or consumed excessive resources

Triton Software (Cork, Ireland)

Period	November 2016 - February 2017
Title	PHP developer

Full stack development on a medical record system used by nearly all cruise ships. Strictly monitored by coast guards: no record can ever be deleted and everything follows comprehensive guidelines. Worked directly with clients on issues and new enhancements.

Apple (Cork, Ireland)

Period	May 2016 – November 2016
Title	QA, Localisation Tester

Testing Apple software products, raising bug tickets and language proofing Dutch translations.

Ordina (Nieuwegein, The Netherlands)

Period September 2015 – May 2016

Title Software Engineer

Rapid prototyping consultant, building experimental prototypes for third-party clients: a model taxi parking system for KLM at Schiphol, a gamified Apple Watch health app for Interpolis, and an offline-capable evacuation app for ProRail (Dutch railroad infrastructure).

EDUCATION

College of Amsterdam 2012 - 2015
BSc. Technical Software Engineering

Nova College 2010 - 2012
MBO Software development level. 4

VLC college 2007 - 2010
HAVO-3

BEYOND THE CODE

- Road cycling with my local club — nothing builds team spirit like a shared headwind
- Growing our own vegetables on the allotment
- Tinkering with electronics: small LoRa, ESP32 and nRF52 side projects

LANGUAGES

Dutch mother language

English spoken: excellent
writing: excellent

COMPETENCIES

- Mentoring — taught teams Kubernetes and decoupled software design
- Fast onboarding — repeatedly became the go-to person on undocumented legacy systems
- Pragmatic modernisation — new technologies into production without disrupting existing workflows
- Ownership — on-call experience, production debugging and security hardening