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	8
Front-End Technologies	9
Azure Devops	8
.Net Core	7
Docker	6
Kubernetes	6

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
- AngularJS
- VueJS

Knowledge Back-End

- C#/.NET - including .NET Core
- Entity Framework
- Elasticsearch
- Various message brokers (Kafka, RabbitMQ)
- Rest API, Middleware
- Microservices and Monoliths
- Git
- Kubernetes

Knowledge DevOps

- ELK/Logging systems
- PostgreSQL, MySQL
- Azure DevOps
- Helm
- Docker
- Prometheus, Grafana
- Serverless architectures

Work Experience

Introduction

I enjoy learning about and gaining experience in different industries and in different company structures, specifically focusing on the tech stack used.
This can greatly benefit a company as I am able to give the ups and downs of any of the technology stacks. I am highly adaptable and well able to adjust to new code. The things that I strive for:

- **FrontEnd:** HTML/CSS/JS, starting off as a means to get a button to do something in the BE, but quickly learned that users need good UI to be able to understand what the data does and in recent years found this "harder" to get right than the backend, a CRUD is easy, doing something cool with Background workers to handle data and process is fun. But if the user doesn't see any of this data, it is in essence useless. It is a fun challenge to get something looking right to the user while still displaying complex concepts that the user might not be able to understand if it wasn't for the UX. Frameworks are very easy to pick up once you have a basic knowledge of JS, of which I have very in-depth experience (fun side projects to build things purely in vanilla JS)
- **BackEnd:** Deep understanding of C#, I like to keep up with the latest lang version, understand why something isn't the best approach while still writing something that the next person can understand, where possible use a minimal amount of Nuget packages that are essential and very well maintained and write code that is just immediately understood. Understand most other languages to a basic extent, like C/C++ VB, Java or Python, just enough to convert it to C#!
- **Maintain:** Create very readable code, while keeping an application optimised for speed and reliability is hard, I can quickly understand code and create new features that fit the narrative of the existing code
- **Create new Features:** Quickly deliver code in an iterative fashion, while unit testing already give small previews and ask for feedback
- **Improve developer experience:** Isolating application logic and create a Domain Driven Design to create a more reliable product by creating more Test Driven Design principles
- **Solution architect:** A keen understanding of existing infrastructure and being able to quickly change it if the business requires it, I have experience with mono, microservices and serverless all using IaC and with security in mind and I very well know what to use when
- **DevOps:** Creating strong CI/CD integrations for reliability in deployment, which will in turn shorten the development cycle

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
Languages	C#, bicep
Frameworks	Kubernetes

TBD

The main tasks I had at Uniphar were:

- TBD

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

I was asked back at Kiwa, not for the project that I was originally hired for, but for help with some of their internal systems, since they needed a bit of TLC. During my time I mainly did maintenance and small features to existing projects. But since the applications were mostly written in .net framework I also decided to help out with a migration to .net core, with some minimal effort I was not only able to remove any dependencies that would stop us from switching, but I also successfully migrated the application. Since I can very quickly learn a new system inside out I was also asked to help and support some older applications and to write up some documentation that was not available

The main tasks I had at Kiwa were:

- Help maintain the applications
- Fix bugs, create new features and find new bugs
- Clean up legacy code, upgrade projects and invent upgrade paths
- Help with invoice proposal
- Understand code that nobody in the company had any knowledge of and come up with solutions to get this code more maintainable

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. It was built locally on portainer instances. Little consideration was given to reliability, security and isolation of business logic. My main task was to get a production-ready application launched. By leveraging my years of experience in various software stacks, I was able to create a version of the application that could be run natively on the cloud while maintaining high observability that would be expected of any modern application. Lastly I removed all manual steps such as the provision of new customer environments and ensuring the application could handle large amounts of 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

We have been tasked by Kiwa to produce a proxy type application to collect and process data for the manure industry, since this system had to be asynchronous, reliable, secure and highly flexible, we chose to build this in containers and with kubernetes, since I had previous experience in this, I was tasked with setting up the basic system and design it. Because of my previous experience we chose to use Kafka as the Pub-Sub system, since this is highly reliable and ES for logging, since this is almost a de facto standard by now. We decided for an API based FrontEnd using a utility based CSS and Web components to keep this part also very flexible and scalable. This was the first time I had to do everything from scratch and learned a lot from this, also new for me was the extended way we used EF, this is something I dealt with in the past, but I did not make myself an expert in. We also created our own OAuth server due to the restraints presented to us, all in all a project where we not only proved 'new' technologies are worth it, but they are also highly flexible and maintainable when done right.

A few selections of the main tasks Daniël did were:

- Build, design and implement the infrastructure
- Introduce new technologies to Kiwa who still runs some of their servers on bare metal and has no prior history with technologies like Docker
- Build, develop and CI/CD backend services to process: SOAP, XML, SFTP and have restful APIs to have a very flexible set of data to insert and collect
- Create a dynamic FrontEnd mostly driven by data entries (tables) that need to be very readable, responsive and flexible for future changes
- Create a full OAuth flow, which has multiple login systems, with roles that can be maintained from the FE, it also has a "Permission Module" uniquely developed for the Agriculture market, but highly customizable for future portals
- Improve security throughout the whole process, create Tunnels using Cloudflare, have E2E HTTPS traffic, create a Zero trust security plan where security standards are higher than any company I ever encountered

Univé (through TeamRockstar) (Zwolle, The Netherlands)

Period	August 2019 – May 2020
Branch	Insurances
Title	Front end Web Developer
Methods & Techniques	Scrum (Jira)
Operating systems	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 there were a lot of older techniques that were being deployed, for example jQuery. This in combination with the hard to read code, made developing slower for all the developers. That is why at Univé I introduced a few technologies, mainly Web-components (using LitElement) in combination with a Utility based CSS called Tailwind. This made developing quicker and more productive, without changing too much of the current workflow, but it dramatically decreased the size of both the JavaScript libraries and the size of CSS, on new components. To make it even better to manage and develop in I also introduced Karma for unit testing and Storybook as a component library, this so UX could view the components in an earlier stage and for us to have a better view of what is where and how. This allowed us to create a more powerful User Experience, while maintaining a consistent look and feel.

A few selections of the main tasks Daniël did were:

- Maintaining unive.nl and the logged in environment of unive.nl
- Introducing new technologies tailored to the product and bringing this to production (Tailwind, LitElement)
- Maintaining build and release pipelines and improving these for quicker builds and better stability
- Create new features, that with the limited availability of other systems, still gave a rich user experience, by finding creative solutions, rather than limiting yourself to the systems you don't have control over
- Being a helping hand for other team members, when they can't figure out how to build new or improve 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)
Operating systems	Azure DevOps, K8s, Docker
DB/DC	Nifi
Languages/Tools	.net core 2.2/K8s/Docker/MassTransit/RabbitMQ

Within RTL there are loads of systems that need to communicate with each other, the main idea of this project is to combine this, going from a few source systems to multiple target systems, thus reducing load. This also makes it possible to integrate legacy systems with the newer systems RTL acquired, this data in the end will serve as metadata for platforms like RTL XL and Videoland

A few selections of the main tasks Daniël did were:

- Adapt/Receive data from 1-2 source systems
- Create Distributors that Distribute these to 4 target systems, each tailored for their target system needs
- Finding and resolving both technical and non-technical queries
- Build a CD/CI pipeline in Azure DevOps
- Create and setup Azure resources
- Guide the team as a team lead, introducing Kanban and getting business requirements from POs

MyDataFactory (through TeamRockstar) (Meppel, The Netherlands)

Period	October 2018 – February 2019
Branch	Matching Data
Title	Full stack C#/Angular developer
Methods & Techniques	Scrum (Jira)
Operating systems	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.

A few selections of the main tasks Daniël did were:

- Building a Front-End where the matches can be suggested and approved, and after this processing, can be downloaded for other systems
- Building and deploying a proof of concept to move all the current infrastructure to Kubernetes with code as infrastructure, in which I used helm to automate this better
- Moving the existing tag structure to a single system, with some minor improvements and building full deep dive tests to guarantee tag behavior currently and in the future
- Move existing Azure functions to pods in Kubernetes, which are now using Kafka to get queue messages, this to improve performance significantly, after which the data will be saved 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)
Operating systems	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). Currently the old stack needs to be phased out since the code is never maintained and is not correctly built. So Daniël and the rest of the development team is working on this new stack in .net Core(v2.1) + angular (v6) to create the same functionality as the old stack, but with many improvements and optimisations. Partly by improving on data flow and partly by cutting cost, utilizing more Serverless approaches like lambda

A few selections of the main tasks Daniël did were:

- Build new front end functionality, improve and refactor existing code in angular and VMware's clarity which interacts with a backend
- Create lambda's that get kinesis events for parsing data from sensors that are in the field, which then get processed and finally stored if all data is correct

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

The fraud prevention system main task was:

- Fact/Score based system to indicate if a transaction that our customer sent us is a legit person
- Get the good guys in, make it very hard/impossible for the bad guys to come in
- Aggregation over big sets of data to determine if the end user is a valid user (identify the user based on their history)

A few selections of the main tasks Daniël did were:

- Integrate with 3rd party data sources for better fact results, for example: xml, soap, custom xml, json etc.
- Extend existing API and service these new and existing API in a VueJS application for demo purposes
- Creating Test systems/Mock systems for better QA automation
- Improve existing delivery pipelines to speed up releases and remove manual interaction
- Building custom nifi processors for handling data streams
- Being on an on call rota for outages (remote logging into systems to determine why the system went down or is using so many resources)

Triton Software (Cork, Ireland)

Period	November 2016 - February 2017
Branch	Medical systems for cruise ships
Title	PHP developer
Methods & Techniques	Scrum
Operating systems	Ubuntu 17.04 LTS
DB/DC	MySQL
Languages/Tools	PHP, HTML5, CSS3, Bootstrap, NetBeans

Combined back-end and front-end developer. I develop a medical record system for nearly all cruise ships, which are strictly monitored by coast guards, no record can ever be deleted and records require to be implemented using comprehensive guidelines.

Daniël worked directly with clients on issues that require development as well as creating completely new enhancements set by the company.

Apple (Cork, Ireland)

Period	May 2016 – November 2016
Branch	iPhone, iPad, Mac, Watches, TV
Title	QA, Localisation Tester
Methods & Techniques	Ticket system
Operating systems	OS X

Testing of Apple software products, logging and raising bug tickets and language proofing of translations in my mother tongue (Dutch).

Ordina (Nieuwegein, The Netherlands)

Period	May 2016 – November 2016
Branch	Smart Technologies
Title	Software Engineer
Methods & Techniques	Scrum, Agile
Operating systems	Windows 7
DB/DC	Couch DB
Languages/Tools	Angular JS, HTML, Bootstrap, Swift, Objective C

As a consultant working in the rapid prototyping department Daniël was responsible for the development and creation of several technical prototypes for automation systems for third party clients. Projects include:

- Model taxi parking system for KLM: Creating an experimental parking system for taxis near the Netherlands biggest airport on a small scale. Removing much needed space while removing waiting time for taxi's, so they can be more efficient.
- Smart health app for Interpolis: Health app for iOS using Apple Watch biometrics designed with heavy gamification to reduce and prevent stress that leads to burnout and its high exposure to health insurers.
- Evacuation application for ProRail: Created a custom evacuation application, that even works when the internet is temporarily not available for Dutch railroad infrastructure.

COURSES

College of Amsterdam 2012 - 2015
BSc. Technical Software Engineering

Nova College 2010 - 2012
MBO Software development level. 4

VLC college 2007 - 2010
HAVO-3

LANGUAGES

Dutch	mother language
English	spoken: excellent writing: excellent

COMPETENCIES

- Passion for learning
- High determination
- Stress resistant
- Result driven
- Team player
- Precise
- Social
- Out of the box thinker