Dragan Dimitrijevic

Software Engineer

Mobile:
+45 26355730
Email:
dragan@visidra.com
Website:
dragandimitrijevic.com
Linkedin::
https://www.linkedin.com/in/ddragan/

About

I'm a software engineer with over 15 years of experience building scalable, backend-oriented web applications. My expertise lies in designing cloud-native architectures and developing robust APIs, developer tooling, and automation workflows-primarily using Node.js, TypeScript, and AWS.

In the last six years, I've specialized in AWS-based application development, working extensively with services like Lambda, EC2, RDS, Step Functions, DynamoDB and API Gateway. I build solutions with a focus on performance, maintainability, and infrastructure-as-code practices using CDK and Serverless Framework.

I enjoy working across the entire development lifecycle-from architecture and integration design to deployment and monitoring. I'm also deeply familiar with SaaS delivery models, CI/CD automation, and multi-tenant systems. Outside of work, I'm passionate about Open Source, staying physically active, and spending time with family and friends.

Work Experience

Senior Backend Developer

Appvestor - Copenhagen *Oct 2023 - Jan 2025*

As a Senior Backend Developer at Appvestor, I was focused on optimizing the Doralytics (reporting and analytics system) and migrating finance.appvestor.com from PostgreSQL to MySQL. My main responsibilities included:

- Downsizing AWS resources and decommissioning obsolete system components.
- Migration of application vitals from Google APIs to AWS Step Functions (SFN) and moving credentials from EFS to MySQL.
- Developing Lambda functions to calculate campaign attribution on EC2 instances.
- Migrating insertion orders from finance.appvestor.com, creating an OpenAPI specification, and implementing an Insertion Orders REST API using Python and Flask.
- Enabling data synchronization from PostgreSQL to MySQL through the aws_lambda extension, executed within PostgreSQL triggers.
- Setting up a notifications Lambda function, triggered directly from MySQL via lambda_sync.

These efforts improved resource efficiency, streamlined data workflows, and strengthened backend system functionality.

Key skills: 📚 AWS, Nodejs, Typescript, Python, MySQL, PostgresSQL

Software Engineer

kompasbank - Copenhagen *Nov 2021 - Mar 2023*

This was my first role in a fintech industry. I worked on an in house Lead Generation application. It is a tool developed primarily for the sales team but it is used throughout the organization. The primary

objective was to profile companies with solid economy who would be suitable candidates for credit approvals. The input from the sales team was fed into the ML model which analyzed annual reports, probability of distress and other financial information relevant in minimizing risks.

Another interesting part of my journey at kompasbank was working on the credit automation project. In order to process large batches of credit request, we designed a workflow solution which integrates external on boarding system (Muinmos) with credit approval REST API and stores the relevant information back into kompasbank's CRM nCino (Salesforce). Once the client information passes KYC, AML and PEP checks, the financial data is being analyzed and the credits are being approved or disapproved.

The software was developed using Nodejs and Typescript for both front end, back end and AWS cloud resource provisioning. For the workflow logic we chose AWS SFN service which proved to be an excellent choice for this problem domain.

Key skills: 📚 Nodejs, Typescript, AWS, CDK, DynamoDB, React, Angular

Software Engineer

Freelancer - Copenhagen *Jan 2020 - Oct 2021*

In early 2020, I began working as a freelance developer — a journey that's been both deeply rewarding and occasionally challenging. This period has allowed me to expand my technical skills, manage projects independently, and work closely with clients to deliver real-world solutions.

One of the most significant projects during this time was building a custom Airship integration for Oracle Eloqua, enabling enterprise-grade push messaging within Eloqua's Campaign Builder.for Eloqua.

Key skills: 📚 Nodejs, AWS, Serverless, Angular, Angular Material

Software Architect - Head of product

Nordlid - Copenhagen *Jan 2018 - Nov 2019*

At Norlid, many clients relied on Oracle Eloqua as their marketing automation platform, and I was responsible for developing a range of custom solutions and integrations tailored to their business needs. These often involved building logic around Eloqua's Contact, Asset, and Campaign systems to support dynamic workflows and extend core functionality.

A large part of my work focused on integrating third-party data sources — importing Contacts and related information, and exporting activity data such as email clicks, web visits, and push responses.

Over time, this experience led to the development of a set of standardized SaaS solutions, installable across multiple Eloqua instances. In my additional role as Head of Product, I was responsible for transforming bespoke solutions into reusable software products, deployed and managed entirely on AWS infrastructure.

Beyond Eloqua work, I also developed the backend and mobile app for Få tiden tilbage, a six-month-

long project focused on productivity and time tracking in the context of railway maintenance and service delays.

Key skills: 📚 Nodejs, AWS, Serverless, Angular, Ionic

CTO

Globase International - Copenhagen *Jul 2016 - Dec 2017*

At Globase, I led a development team of five engineers and helped support a client portfolio of over 100 companies, including Mercedes-Benz, Bang & Olufsen, 3M, Grundfos, DFDS, and Cognito. The core product was a SaaS Email Marketing Automation platform, complemented by a suite of custom solutions, microsites, and landing pages.

My responsibilities included:

- Leading the development team using Agile SCRUM practices
- Maintaining and evolving internal platforms and client-specific solutions
- Participating in pre-sales and requirements specification
- Managing physical server infrastructure hosted in QSC (Germany)
- Serving as a member of the company's management group
- Overseeing the migration of clients from the legacy V1 platform to the modernized V2

This was a dynamic period marked by technical transformation and cultural shifts, as Globase transitioned from Ad Pepper Media ownership to MailUp (Growens).

Key skills: 📚 Linux, Apache, MySQL, PHP, Nodejs, Express, ActiveMQ, Vagrant, Ansible

System Developer

Increase - Copenhagen *Nov 2015 - Jun 2016*

During my time at Increase, I primarily worked with Oracle's Eloqua platform, building custom applications and data integrations in the AWS cloud.

One of the more notable projects was the development of the Event App — a solution designed to automate and manage event workflows within Eloqua. The app dynamically created campaign assets such as emails, landing pages, registration forms, and custom data objects. It supported the full event lifecycle, including invitation, registration, and real-time participation tracking.

Key skills: 📚 Nodejs, AWS, Sails, Express, Jquery, HTML5 / CSS3

System Developer

Globase - Copenhagen *Apr 2014 - Oct 2015*

I developed custom solutions and integrations for an Email Marketing Automation platform. This included tasks such as integrating external data sources for dynamic email content and implementing interactive campaign strategies like polls, quizzes, and giveaways.

Key skills: 📚 PHP, Javascrip, Laravel, MySQL, Linux

Software Developer

Brandhouse - Copenhagen / Odense *Mar 2013 - Mar 2014*

At Brandhouse, I had the opportunity to work on Publicator, a platform designed for graphic design and print. My responsibilities included integrating external data sources, implementing new feature requests, and gathering requirements directly from customers.

Key skills: PHP, Apache, MySQL, Linux, JavaScript, HTML5 / CSS3

Developer

Peytz & Co - Copenhagen *Sep 2011 - Feb 2013*

This was my first role in the marketing industry, where I gained hands-on experience with an email automation platform. I developed new features and custom solutions that enabled users to integrate content from external sources, including SOAP APIs, REST APIs, and XML feeds.

Key skills: PHP, Apache, MySQL, Linux, JavaScript, Jquery, HTML5 / CSS3

Developer

MOCH - Copenhagen *Mar 2007 - Aug 2011*

During my time at MOCH, I worked on a modular, SCORM-compliant Learning Management System (LMS) platform. My responsibilities included developing feature-rich modules such as LMS courses, workflows, and questionnaires, as well as implementing new functionality and resolving technical issues.

Key skills: PHP, Apache, MySQL, Linux, JavaScript, HTML5 / CSS3

Developer

Tang Data - Lejre *Jun 2006 - Feb 2007*

My first professional experience involved developing an in-house ERP platform for veterinarians. I focused on migrating functionality from an older version, implementing new features, and troubleshooting reported issues.

Key skills: C++, Qt, Windows, PostgreSQL

Projects

TimeTracks

A lightweight, headless, developer-centric time tracking platform designed to support individual users, teams, and automated workflows like CI/CD pipelines.

- Built a full-stack web application with Node.js (Express), DynamoDB on AWS (API Gateway, Lambda, CDK)
- Implemented multi-tenant support, JWT-based authentication, and token-scoped API access for

secure, scalable usage
- Designed the system to support both manual time entry and automated tracking for background jobs and deployments
- Developed OpenAPI-based documentation and integrated Redocly for developer-friendly API references

Launched open beta with early user adoption and a lifetime-free offer for early signups

Tech stack: 🛠️ Nodejs, Express, AWS CDK

Cloud services: ☁️ Lambda, DynamoDB, API Gateway, CloudFront, CloudMetrics

SMS Cloud Connector

Since Eloqua primarily focuses on email marketing and campaign management, it does not include SMS messaging as a native feature. To address this limitation, I partnered with my client to develop a custom Eloqua application that integrates with three major SMS providers:

- Inmobile
- SMSDK
- Sinch

The integration was implemented as an SMS Cloud Connector, allowing marketers to drag a component directly onto the Eloqua campaign canvas. With just a few clicks, users could select the appropriate contact field and design dynamic SMS message templates.

A standout feature of this solution was the integration of the Mustache templating engine, which enabled advanced personalization by merging dynamic data—including stringified JSON fields—directly into SMS bodies.

Message delivery status was tracked through a callback-based reporting mechanism, allowing contacts to be flagged as successful or failed based on delivery outcomes. This data was then synced back into Eloqua for tracking and analytics.

Overall, the integration streamlined SMS functionality within Eloqua, giving marketers a seamless, scalable way to incorporate SMS into their multi-channel campaign strategies.

Tech stack: 🛠️ Nodejs, Angular, serverless

Cloud services: ☁️ Lambda, RDS, SQS, S3, VPC, API Gateway, CloudFront, CloudMetrics

Airship Integration

This integration project enabled Oracle Eloqua to send push notifications via Airship,a multi-channel messaging platform. The solution supported several key use cases:

- Sending push notifications from the campaign canvas (scheduled or in-app)
- Triggering batch push messages from Eloqua segments
- Sending push notifications on form submissions
- Importing mobile device data into Eloqua

- Tracking and managing push response activity

Within the Eloqua campaign canvas, marketers could easily drag the Airship Cloud Action component into their workflows. Through a simple configuration process, users could authenticate, select a target mobile app, choose a message template from Airship, and specify platform targeting (iOS or Android). Push notifications could be sent instantly or scheduled, providing both flexibility and ease of use.

The integration also supported batch segment pushes. From the segment view, users could define segmentation criteria and configure push delivery via the embedded Airship interface. While the configuration process was similar to canvas-based sends, the underlying mechanism differed:

- In the campaign canvas, contact payloads were delivered directly to the backend as users flowed through the workflow
- In segment-based sends, contact records were fetched from Eloqua in bulk

Additionally, the integration enabled device imports into Eloqua, storing device data in Custom Data Objects, and logging user interactions (e.g., opens, taps) as External Activities. These data points supported advanced segmentation and enabled more sophisticated user engagement strategies.

Tech stack: 🛠️ Nodejs, Angular, serverless

Cloud services: ☁️ Lambda, RDS, SQS, S3, VPC, API Gateway, CloudFront, CloudMetrics

Få tiden tilbage

Få tiden tilbage was a six-month project developed for DSB, the Danish public transportation company. The concept focused on compensating passengers affected by railway maintenance delays by returning time as a form of value.

In the initial release, the app allowed passengers to check in at the start of their journey and check out at their destination. Based on predefined rules for each route, users earned “minutes” — a form of credit — for completed trips. These credits accumulated over time, representing the value of delays experienced.

Once users reached a certain credit threshold, they could redeem their minutes for in-app ticket purchases. The second release introduced a route selection and ticketing system: users could choose from eligible destinations, redeem credits for a QR-code ticket, and store it for future travel. These tickets were fully compatible with DSB’s on-board scanners for validation.

The project was recognized with a Danish Digital Award in the “Digital Activation” category, acknowledging its innovation and positive public impact.

Tech stack: 🛠️ Nodejs, Ionic, serverless

Cloud services: ☁️ Lambda, RDS, S3, VPC, API Gateway, CloudFront

Rich Relevance

The objective of this project was to integrate the Rich Relevance Customer Experience and Personalization Platform into Oracle Eloqua as a standard application. The integration leveraged Rich

Relevance’s Recommend and Engage products to deliver personalized content within Eloqua email campaigns.

Personalization was achieved using Eloqua’s Cloud Content feature, allowing marketers to drag and drop dynamic content blocks directly into email templates. This setup enabled personalized content to be generated at the moment when email content renders, based on predefined logic.

Two types of personalized content were supported:

Recommend

The Recommend integration allowed marketers to configure personalized product suggestions. Through a user interface, they could search for products or categories using data pulled from Eloqua Custom Data Objects (CDOs). Visual presentation templates were also managed within Eloqua and made available via dropdown selectors. These templates, combined with selected product data, were used to construct the necessary parameters for dynamic content rendering within emails.

Engage

The Engage product followed a similar configuration flow but focused on serving personalized banners, ads, and promotional content. This type of content was simpler to manage, offering quick ways for marketers to enhance visual impact and user engagement within their campaigns.

Tech stack: 🛠️ Nodejs, Angular, serverless

Cloud services: ☁️ Lambda, RDS, S3, VPC, API Gateway, CloudFront

Event App

The primary goal of this project was to build an application that simplifies and streamlines event management in Oracle Eloqua. Creating event flows manually in Eloqua is often repetitive and time-consuming. This application was designed to automate that process — enabling marketers to set up event logic and required assets with minimal manual effort.

During the event creation process, the application automatically generates and configures Eloqua assets such as Segments, Emails, Forms, Landing Pages, Campaigns, and Custom Data Objects (CDOs). These assets are then integrated into the Eloqua campaign canvas to support the full event lifecycle, including invitations, registrations, cancellations, and waiting list handling.

Once a campaign is activated, Contacts in the segment receive an invitation email. By clicking the registration link, they enter the event flow. Participant status (e.g. registered, attended, unregistered, waiting) can be updated through the application and is tracked via CDOs using Eloqua processing steps.

To maintain consistent branding and reduce setup time, the application supports reusable Asset Templates for emails and landing pages. This enables high levels of customization and reusability across events.

Tech stack: 🛠️ Nodejs, jQuery, Sails, Docker

Cloud services: ☁️ ECS, RDS, VPC, CloudFront

Skills

General:

AWS, Nodejs, JavaScript, Typescript, PHP, Python, MySQL, Postgres, DynamoDB, Linux, Git

Frameworks:

Angular, Angular Material, React, MUI, Ionic, Express, Sails, TypeORM, Sequelize, Jest, Laravel, Codeigniter, Jekyll

CI/CD & Cloud Tools:

Serverless, AWS SAM, AWS CDK

AWS Services:

EC2, ECS (Fargate), Lambda, SFN, RDS, DynamoDB, S3, Cognito, API Gateway, AppSync, CloudFront, CloudWatch, VPC

Education

Engineering College of Copenhagen (IHK), Information and Communication Technology *2003 - 2006*

Niels Brock, Datamatician *2001 - 2003*

K.I.S.S, DU3 Modul 5 *2004 - 2005*