

DRAGAN DIMITRIJEVIC
CURRICULUM VITAE

About

Personal Information



Dragan Dimitrijevic

✉ dragan@visidra.com

☎ +45 26355730

🇷🇸 Serbian

🇩🇰 Danish

🎂 30.11.1978

Introduction

I am a Software Engineer with over 15 years of experience in application development and architecture. I have started with the LAMP stack many years ago and as my interest for the JavaScript and frontend development grew, I moved slowly over to Nodejs and later onto cloud computing with AWS.

I have been developing software in AWS for about 6 years now, working primarily with services such as Lambda, EC2, ECS, RDS, API Gateway, Cognito and SFN. I enjoy working on cloud based solutions as they offer scalability and a wide variety of services which can be used to build feature rich applications.

I have a great deal of interest for Open Source technologies. Integrations and application development are the areas in which I spent most of my professional life. My recent technology stack is Nodejs and Typescript, Angular and AWS. In my spare time I enjoy training, running and spending time with my friends and family.

Languages

- Serbian (native)
- English
- Danish

Work Experiences

Software Engineer

kompasbank

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.

Software Engineer

Freelancer

Jan 2020 - Okt 2021

In January 2020 I decided to embark on a freelance journey. Its been a great learning experince and a bit challenging at the same time. One of the biggest project during this time was developing Airship integration for Eloqua.

Software Architect - Head of product

Nordlid

Jan 2018 - Nov 2019

Since many of the Norlid's clients use Eloqua as an Email Marketing Automation platform, I was working on developing custom solutions and integrations. Most of the solutions would typically involve a specific business logic added on top of Eloqua, which would handle Contacts, Email Content or manage Assets in campaign canvas. Integrations were concerning primarily with importing Contacts and related data from different sources, and exporting data on Contact activities such as email clicks, web visits, push response etc.

As a result of many years of developing software around Eloqua, a series of products have emerged as standard packages. These SaaS type solutions were designed to be installed across instances and serve the needs of multiple clients. With my second role as a 'Head of product', I had a responsibility to create standard software based on the experiences from custom solutions. All the applications have been developed and hosted on AWS cloud.

Besides building software for Eloqua, I had an opportunity to develop a backend and mobile application for "Få tiden tilbage" project which took six months to complete.

CTO

Globase International

Jul 2016 - Dec 2017

At Globase, I was responsible for a development team of 5 employees. The company had a client portfolio of about 100 companies, among which were: Mercedes - Benz , Bang & Olufsen, 3M, Grundfos, DFDS, Cognito and others. Our SaaS product was an Email Marketing Automation platform with an eco system of custom solutions, microsites and landing pages.

List of responsibilities:

- Managing development team (Agile SCRUM)
- Maintaining in house platforms and custom solutions

- Pre-sales and requirement specification
- Maintaining server racks in QSC (Germany)
- Being part of the Management group
- Migrating clients from V1 to V2 platform

It was an exciting time of transition and cultural changes due to the fact that Globase changed the ownership, from Ad Pepper Media to Mailup (Growens).

System Developer

Increase

Nov 2015 - Jun 2016

Most of my time at Increase I spent working with Oracle's Email Marketing Automation platform Eloqua. I was building applications and data integrations in the AWS cloud. One of the more interesting project during that time, was the development of the Event App. This application was designed to handle the event flow by creating many assets (Email, Landing Page, Registration Form, Custom Data Object etc) on a campaign canvas in Eloqua, which can handle the invitation, registration and keep track of the participation status.

System Developer

Globase

Apr 2014 - Oct 2015

I was developing custom solutions and integrations against an Email Marketing Automation platform. This involved a variety of tasks such as integrating external data sources for email content, implementing contest strategies such as polls, quiz and giveaways.

Software Developer

Brandhouse

Mar 2013 - Mar 2014

At Brandhouse, I had a chance of working on a platform (Publicator) made for graphic design and print. The tasks were including integration with data sources, implementing new feature requests and collecting requirement specification from a customer.

Developer

Peytz & Co

Sep 2011 - Feb 2013

This was my first job in the Marketing industry where I had an opportunity to work with an email automation platform. I developed new features and custom solutions which would allow users to integrate content from external sources such as SOAP API, REST API, XML Feeds etc.

Developer

MOCH

Mar 2007 - Aug 2011

During my time at MOCH, I worked on a modular SCORM compliant LMS (Learning Management System) platform. My tasks were development of modules (LMS Courses, Workflow, Questionnaire), adding features and fixing issues.

Developer

Tang Data

Jul 2006 - Feb 2007

My first work experience was developing on an in house ERP platform designed for veterinarians. I worked primarily on migrating from an old version, adding new features and troubleshooting reported problems.

Projects

Airship

This integration project was about making Eloqua send push messages, using multi channel messaging platform [Airship](#). The integration was covering following use cases:

- Send push message from campaign canvas (scheduled, in-app)
- Send push message from a segment (batch)
- Send push message on form post
- Import devices to Eloqua
- Collect response activity

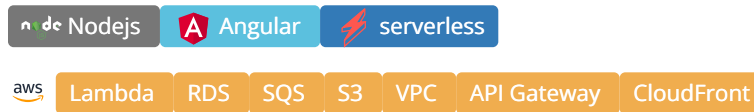
When working in campaign canvas view, a marketer can choose an Airship component (cloud action) and drag it onto the campaign canvas. In just a few clicks, a marketer can authenticate, choose a mobile app, select a message template (configured in Airship) and choose Android or iOS as target platforms. If necessary, a message could be scheduled or send as an immediate push. Once everything is saved, the campaign is ready to send push message to all the contacts that pass through the action step.

Besides sending via campaign canvas, a marketer can choose to send push message to a whole segment in a segment view. Once the criteria for the segmentation is defined and saved, the Airship integrated GUI can be slided out and the configuration steps can take place. As in the previous scenario, the configuration process is the same, however behind the scenes the difference is that segment push is a one time send out to all the segment members, while in the ongoing campaigns the messages are send to passing contacts. The big difference here is that in the first scenario, the integration backend receives contact payload, whereas in the second the contact payload has to be fetched from Eloqua.

Apart from sending push messages, the Airship integration also manages the import of Airship devices into Eloqua and collects response activity. Devices are stored in Custom Data Objects, while

response data are stored as External Activities. Both are used in segmentation process and in order to create more complex flows based on user interaction.

Techstack



Få tiden tilbage

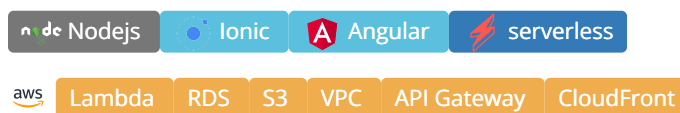
“Få tiden tilbage” was an interesting projects that took almost 6 month to complete. The client was a public transportation company (DSB) and the concept was focusing on bringing back value to the passengers who experience delays, due to the maintenance work done on the railways.

The first release of the application was managing journeys, where a passenger is able to check in on the start of the journey, and checkout on the end destination, earning a certain amount of minutes based on the rules defined for that specific route. Every journey route had a defined award, and every journey was adding to the total of the credits saved.

Once the saved credits reached over a certain limit, the users were able to use those credits to make an in-app ticket purchase. In the second release of the application, users where able to choose from a specific set of destinations, select a route that they can afford, purchase and save the ticket for later use. The tickets with the QR codes where complied to DSB scanners so that they could be checked for validation while traveling in the trains.

For this project, Nordlid has recieved the Danish Digital Award in the category “[Digital Activation](#)”

Techstack



Rich Relevance

The objective of this project was to integrate Rich Relevance Customer Experience and Personalization Platform as a standard Eloqua application. The content personalization was implemented using products Recommend and Engage. The main objective was to integrate personalized content into Email Templates by using 'Cloud Content' drag'n'drop feature in Eloqua. This would enable marketer to configure a cloud service so that the content can be generated dynamically as email gets opened.

There were two types of content types that could be integrated, which are related products and content promotion.

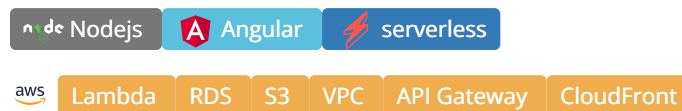
Recommend

Recommend content was handled in such way that the marketer could search for a specific product (or category) in the user interface, while pulling data from the CDO in the Eloqua. The layout for the visual presentation are also placed in Eloqua select drop downs. This allows a marketer to configure elements in the Eloqua instance and then use them in the Rich Relevance integration in order to construct necessary parameters which are used during the Email rendering.

Engage

Engage content is handled and configured in the similar way, it is slightly simpler to work with since it shows advertisement, banners and similar content.

Techstack



Event App

The main objective of this project was to create an application which would help facilitate event management in Eloqua. In many cases managing events in Eloqua can be a repetitive task, requiring many hours of work in order to setup the necessary logic for an event flow.

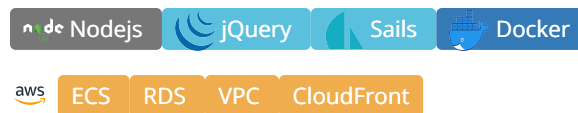
This application attempts to do exactly that, save the marketer from manual work and create the assets and the pattern for the participation flow.

During the event creation process, many Eloqua assets are being created and integrated into the campaign canvas. Assets such as Segment, Email, Form, Landing Page, Custom Data Objects and Campaign are created and configured in order to support invitation, registration, unregistration and waiting list email notifications. When a Segment is ready and the Campaign activated on the campaign canvas, the Contacts on that segment would receive an invitation email and by clicking on the registration link, enter the event flow.

A participant can have different status changes such as registered, attended, unregistered and waiting. All these status changes can be initiated within the application and stored in the Custom Data Object in Eloqua. The change of participant status (registered, attended, waiting) are handled via processing steps.

In order to achieve a specific look and feel for the Emails and Landing Pages, Event application supports Assets Templates. This allows a high level of customization and re-usability.

Techstack



Education

Information and Communication Technology

Engineering College of Copenhagen (IHK)

2003 - 2006

DU3 Modul 5 - Prøve i Dansk 3

K.I.S.S

2004 - 2005

Datamatician - Information Technology

Niels Brock

2001 - 2003

Skills

General

Nodejs



JavaScript



Typescript



Linux



PHP



MySQL



Apache



Qt/C++



Ruby



HTML5/CSS3



Git



Frameworks

Express



Serverless



CDK



React



Angular



Angular Material



Ionic



Sails



TypeORM



Sequelize



Jest



Laravel



Codeigniter



Jekyll



AWS

EC2



ECS



Lambda



SFN



RDS



DynamoDB



Cognito



API Gateway



AppSync



S3



CloudFront



CloudWatch



VPC

