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. My journey began with the LAMP stack, but as my interest for JavaScript and frontend development grew, I transitioned to Node.js. Around the same time, cloud computing became the industry standard, leading me to focus on development and deployment with AWS.

For over six years, I have been developing software on AWS, working primarily with services such as Lambda, EC2, ECS, RDS, API Gateway, Cognito, and Step Functions. I enjoy building cloud-based solutions because they offer scalability and a rich ecosystem of services, enabling the development of feature-rich applications.

I have a strong passion for Open Source technologies and have spent most of my career focused on integrations and application development. My current technology stack includes Node.js, TypeScript, and AWS. In my spare time, I enjoy training, running, and spending time with friends and family.

Languages

- Serbian (native)
- English
- Danish

Work Experiences

Senior Backend Developer

Appvestor

Okt 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.

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 challending 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 where 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 where 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 responslible for a development team of 5 employees. The company had a client portfolio of about 100 companies, among which where: 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 where 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 a 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

SMS Cloud Connector

Since Eloqua is primarily focused on email marketing and campaign management, SMS messaages are not included as a core, native feature. To address this, I developed a custom Eloqua application that integrates with three major SMS service providers:

- Inmobile
- SMSDK
- Sinch

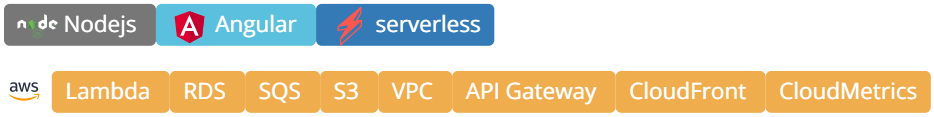
In the Eloqua campaign canvas, marketers can easily select an SMS Cloud Connector component and drag it onto the campaign workflow. With just a few clicks, they can choose the contact field to merge and design a dynamic message template.

A key feature of this project was the integration of the Mustache templating engine. This was particularly valuable for Eloqua clients who had stringified JSON fields that needed to be dynamically merged into the SMS body. Mustache's templating capabilities made personalization and dynamic content integration seamless and efficient.

The reporting of the message status was implemented via callback method. Contacts were flagged as successful or failed based on the message delivery outcome and then imported back into Eloqua for tracking and reporting.

This integration streamlined the process of adding SMS to Eloqua campaigns, providing marketers with a simple, effective way to incorporate SMS messaging as part of their multi-channel strategies.

Techstack



Airship

This integration project enabled Eloqua to send push notifications through the multi-channel messaging platform Airship . The integration supported several key use cases:

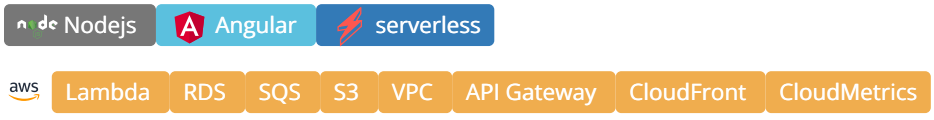
- Send push message from campaign canvas (scheduled or in-app)
- Send push message from a segment (batch send)
- Send push message on form post
- Import devices to Eloqua
- Collecting and managing response activity

In the Eloqua campaign canvas, marketers could easily drag and drop the Airship Cloud Action component into their campaign workflow. With a few simple steps, users could authenticate, select a mobile app, choose a pre-configured message template from Airship, and target either Android or iOS platforms. Push messages could be scheduled for later or sent immediately to all contacts that flowed through the action step, providing flexibility and ease of use.

Appart from the campaign canvas, marketers can also send push notifications to an entire segment. In the segment view, after defining the segmentation criteria, the integrated Airship interface can be accessed to complete the configuration. Although the setup process is similar, there is a key difference: segment-based pushes are one-time sends to all members of a segment, while in ongoing campaigns, notifications are sent individually as contacts pass through the action step. Technically, this means that in the campaign canvas, the integration backend receives the contact payload directly, whereas for segment pushes, contact data must be fetched from Eloqua.

In addition to sending push messages, the Airship integration supports device import into Eloqua and response activity tracking. Imported devices are stored in Custom Data Objects, while response data is recorded as External Activities. Both data types are used in segmentation and help create more complex workflows based on user interactions.

Techstack



Få tiden tilbage

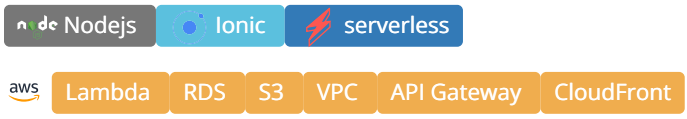
“Få tiden tilbage” was an interesting projects that took almost 6 month to complete. The client was a public transportation company (DSB) and the concept was focusing on bringing back value to the passengers who experience delays, due to the maintenance work done on the railways.

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 marketeer to configure a cloud service so that the content can by generated dynamicaly as email get opened.

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

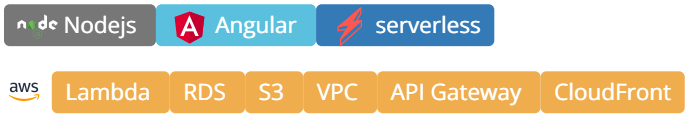
Recommend

Recommend content was handled in such way that the markeeter could search for a specific product (or category) in the user interfrace, 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 marketeer to configure elements in the Eloqua instance and then used 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 marketeer from manual work and create the assets and the pattern for the participation flow.

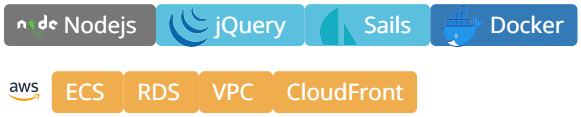
During the event creation process, many Eloqua assets are being created and integrated into the campaign canvas. Assets such as Segment, Email, Form, Landing Page, Custom Data Objects and Campaign are created and configured in order to support invitation, registration, unregistration and waiting list 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

AWS

Nodejs

JavaScript

Typescript

Linux

MySQL

Postgres

PHP

Python

HTML5

CSS3

Git

Frameworks

Angular

Angular Material

React

MUI

Ionic

Express

Sails

TypeORM

Sequelize

Jest

Laravel

Codeigniter

Jekyll

CI / CD

Serverless

AWS SAM

CDK

AWS

EC2

ECS

Lambda

SFN

RDS

DynamoDB

Cognito

API Gateway

AppSync

S3

CloudFront

CloudWatch

VPC