Anders Beck – Detailed Project Descriptions

Backend Developer - Developer Platform
Biotech, Q4 2022 - Q4 2022

The client has had a longer journey to transform from working with project-based deliveries to a product-based culture. This has led to insourcing of software engineering capabilities, establishing of best-practices, ways-of-working and a uniform technical stack. Part of this initiative has been to build a development portal – “by engineers for engineers”. The vision of the development portal is to make it easier for developers to deploy and test their products without interfering with devops.

Anders was brought in as a backend developer to overhaul the backend for the platform as the work had previously been led by a Data Engineer, with limited experience in modern Javascript-frameworks. Anders ported over the old backend to NestJS, it was chosen due to having better maintainability and many out of the box functionalities. Typescript was also adopted in the project, as it previously was not present. To keep the project well documented, OpenAPI and Swagger was used to help future developers get an understanding of the backend API. Lastly, Redis was also adopted in the project as it would serve for caching and queuing of heavy tasks. Anders, together with another Netlighter, managed to get the platform to a state where the client can host projects internally, and also extend the development platform with more ease.

Keywords: Nest.js, TypeScript, Github workflows, OpenAPI, Swagger, PostGreSQL, Prisma ORM, Redis, ArgoCD, Docker

Full Stack Developer - Product Platform
Compliance SaaS Provider, Q2 2022 - Q4 2022

This client builds compliance software that can be tailor made to any company's procedure.

One of the many features of this compliance platform is third party screenings. It allows the customer to conduct screening of third parties such as people or companies that they would want to conduct business with. This has been apart of the clients old platform, but as they move more and more into the new modular engine these third party screenings had to be re written from scratch to fit into the new engine.

As a full stack developer Anders worked both on front- and backend of the clients new platform rebuilding the third party screenings to fit into the modular system. This was quite a challenge as there are many moving parts and things had to be modular so that third party screenings could be used anywhere on the platform. Anders helped achieve this by leveraging his strong engineering skills and ability to adapt to a complex environment. He helped build the parts of engine that allowed the screenings to become modular and refactoring old code to ease development going forward.

Keywords: Node.js, TypeScript, JavaScript, Angular, PostGreSQL, npm

App developer - Integration
Health and Fitness, Q1 2022 - Q1 2022

This client is developing a platform to connect fitness coaches with users, enabling personal and detailed online coaching. This enables users to be matched with a suitable coach, as the selection of coaches is much larger compared to the traditional way. At the same time it enables coaches to scale their business. The clients main focus is currently on physical activity, but the platform is being developed with a view to other areas of focus, such as mental health, as well.

To make their platform even more attractive to both coaches and users, the client wanted data from widely used devices (Apple Watch, FitBit, etc.) to be integrated into their app. This would make it easier for coaches to follow their users at a more detailed level, while making it easier for users to follow their own training and how it matches up against their plan assigned by their coach and even give the possibility to automatically add their activities such as workouts, steps and sleep to mention a few.

Anders' tasks was to investigate, plan and implement how the integration of external data should be done. The client builds their app in React Native using Expo and with a backend developed with Node.js, GraphQL and PostgreSQL. For the implementation, packages that supported this were used, and the team identified Apple HealthKit and Google Fit as the priority platforms that were to be integrated into the client's platform. This was because these platforms allowed the integration of a wide variety of devices, and it would cover both iOS and Android.

Implementation of UI, functionality, and queries to the backend/database were all considered a part of Anders' main responsibilities on the project.

Keywords: React Native, Expo, Node.js, TypeScript, GraphQL, Apple HealthKit, Google Fit, Android, iOS

Full Stack Developer, DEV-OPS - Performance and compliance dashboard

Large retailer, Q2 2020 - Q4 2021

This client is a one of the biggest retailers in the world with operation across the globe in more than 50 countries and with more than 10 million visitors online per day.

The client's website is built on a micro-frontend framework. This enabled many teams to deliver bits and pieces that comes together to create the full experience. While this enables independent and fast development, it does make it hard to keep track of ownership. This among other things had let to poor performance on the website.

To solve this it was decided to create a dashboard to monitor performance and compliance and give an easy overview that anyone could understand and empower the teams to do better.

Anders was part of this project from the very beginning, taking part in scoping what it should be and taking decisions on the architecture. As a full stack developer Anders worked on both the front and backend of the project. The backend consisted of a crawler that collected perfomance and compliance metrics with a mix of off the shelf and tailor made components.

One of these was made to find ownership of all elements (html, javascript, images etc) on the client's website, this was a challenge to do since there didn't exist any data on who owned what. To solve this Anders and his team reverse engineered the upload service to backtrack a file to its upload and map that to a team.

Another component ran Lighthouse tests on several pages and used the result to extract different performance metrics, one of these that was in focus was Core Web Vitals (CWV) which itself consist of three metrics. By extracting this data, making it easily accessible to all teams, and pointing out areas to improve, the client was able to turn all their CWV from being in the red to being green within six months.

Anders worked on every part of this system, from the backend crawler to the frontend that view the results.

Anders had the responsibility to setup this project in GCP and build an automatic pipeline to deploy it. A mix of GCP services was used, for instance Cloud Run was used for both the back- and frontend, Cloud Scheduler was used to trigger scheduled jobs, Cloud SQL was used to host a PostgreSQL DB, Cloud Build was used to automatically build and deploy the project and even setup and destroy test environments for every branch.

The result of this meant that Anders and his team could focus most of their time on the actual code instead of worrying about deployment and cloud setup.

Keywords: GCP, Google Cloud Platform, DevOps, Cloud Services, Node.js, JavaScript, Vue.js, React, Next.js, Terraform, Microservices

Full Stack Developer - Cookie Consent
Large retailer, Q1 2020 - Q1 2021

This client is a one of the biggest retailers in the world with operation across the globe in more than 50 countries and with more than 10 million visitors online per day.

In order to be GDPR compliant the client needed to implement a cookie consent solution. The client settled on OneTrust as a third party provider, which then needed to be implemented onto their website and rolled out across Europe and later the rest of their markets. Together with this the client also wanted to have some internal support and documentation to enable other teams to adopt the solution.

Anders joined this project almost at the beginning. At that time they had set up the third party solution and rolled it out in a few markets. The implementation was build quickly without thinking of scaleability, and as the former developer had left the team, it was now Anders' responsibility to improve the implementation and contunie the roll out for the rest of the markets. With the third party cookie solution constantly changing, Anders' discovered that the implementation needed to be reworked from the ground up to be able to better adapt to the changes from the third party. Anders' reworked implementation proved itself to be a lot more stable and made the rollout go a lot smoother.

Anders also helped shape the way the team worked together, introducing things such a daily standups, propper use of Kanban boards and sprints into the team. This helped the team work more efficiently and be able to divide the work as the team later grew with more developers.

Keywords: Cookie Consent, Micro Frontends, Node.js, JavaScript, Sass, CSS, HTML, OneTrust, Webpack, Bash, Github Action, Edge Side Include (ESI)

Front-end Developer, DEV-OPS - CollabLight
Recruitment, Q3 2019 - Q4 2019

The recruitment department at the client wanted a tool to visualize their internal goals and to make them more motivated to finish them. To do this they wanted to create a sort of game where they could keep track of their goals while competing with another office and getting rewards for doing their work.

A team of 5 consultants (4 developers and 1 project leader / scrum master) was set in to solve the task. A plan was settled which included wrapping the internal goal-tracking functionality in a game to enable the recruitment departments of different offices throughout Europe to compete against each other.

As a front end developer Anders took part in developing new features and improving old ones for the front-end. He was part of the decision making process with stakeholders as to which features to work on and how to form them. Together Anders and the team made an MVP of the game that got deployed into production and used by the recruitment team, which received a lot of positive feedback.

As a dev-ops Anders was a key person in finding the right services on AWS to deploy both front- and backend. He used his knowledge to get both front- and backend up and running on AWS using a mix of different services and creating a simple deploying mechanism.

Keywords: React, JavaScript, AWS, AWS Elastic Beanstalk, Amazon Route 53, Amazon S3 bucket, RESTful API, UX, Node, GitLab, Scrum, HTML

Full Stack Developer - Data Science Lab
Fintech/Banking, Q4 2018 - Q3 2019

As Artificial Intelligence is growing from the laboratory to the market, the client wanted a way to evaluate all these new tools and methods in-house. Therefore, the client created its own Data Science Lab (DSL), DSL will act as a catalyst for machine learning and wider AI initiatives. The DSL aims to do PoC implementations to investigate and demonstrate any potential business value.

Anders joined the team of Data Scientists as full-stack developer to support the DSL with building an MVP for a trade surveillance system powered by machine learning. The team built a micro service responsible for extracting and aggregating the data and providing it over a REST API to be consumed by the machine learning models. Anders also took part in getting the new and existing code production ready by restructuring the code, adding test and bug fixing. Anders was also part of setting up continuous integration and finally the first production deployment. With Anders help the team successfully released the MVP into production with a highly automated solution so keeping it running will be a breeze.

Keywords: Python, Bamboo, Docker, Flask, Python Unittest, DevOps

Full Stack Developer - Twitter bot for custom image classification
IBM, Q2 2018 - Q2 2018

The PR department wanted to develop something that could show of the possibilities with IBM Watson, Anders suggested to create a twitter bot that users could tweet at with an attached image. The bot would then reply to that tweet with the result of an image classification. This idea was very well received and Anders was in charge of the development.

The twitter bot was created in Python and deployed in IBM Cloud as a Cloud Foundry service. It was connected to a custom image classifier made with Watson Visual Recognition, where for each class, a collection of images was made, sorted, uploaded and trained.

Beside his technical responsibilities Anders also had to communicate with other non technical stakeholders in the project and managing a small team for the manual sorting of the images for classification. Using his communication skills, he kept the team informed of his progress and made them understand what could and couldn't be achieved, managing the team he made sure to make the best use of the time so he could work on the development while the team did the manual sorting. This resulted in a highly satisfied PR department, not only because of the product developed, but also due to being kept in the loop throughout the process.

Keywords: IBM Cloud, Cloud Foundry, Visual Recognition, Python, Twitter API, Web Crawler

Full Stack Developer - ARTificial Intelligence FRANK by Cecilie Waagner Falkenstrøm
IBM, Q1 2018 - Q2 2018

Cecilie Waagner Falkenstrøm is a young award winning artist employing new media (e.g. artificial intelligence technology) to create interactive artworks. Her artworks have been exhibited internationally. Cecilie works in the intersection between artificial intelligence (AI), art and philosophy. In her newest art exhibition in London she wanted to create an experience where a person could interact with her "AI" FRANK. Her vision was that when people talked with FRANK, his response would change based on the tone person talking.

The solution was to combine four of IBM's Watson services: Speech to Text, Tone Analyzer, Conversation and Text to Speech. The user would talk into a microphone, then Speech to Text would convert it to text, the Tone Analyzer would analyze the text and its result would go into Conversation. Based on the input Conversation would provide the next part of the chat, which would be said out loud using Text to Speech.

The whole thing was developed using node.js and the APIs for each of the Watson services. It was deployed as a local server running on a Raspberry Pi. While involved in every aspect of the project Anders' main responsibility was to develop the Tone Analyzer module and intergrade it into the Conversation module and to deploy the node.js server to the Raspberry Pi. The project was successful and highly acknowledged for its usability and the well- functioning integration of several Watson services. The appreciation was highly due to Anders technical skills and his ability to comprehend the many different aspects of the project without losing sight of the end-goal.

Keywords: Node.js, IBM Cloud, Watson Conversation, Watson Tone Analyzer, Speech to Text, Text to Speech, Raspberry Pi, Art Exhibitions

Back-end Developer - Calling 1000 people at once
IBM, Q3 2017 - Q3 2017

At the end of IBM Watson Summit Denmark 2017 Cecilie Waagner Falkenstrøm had a performance where she talks to her "AI" persona FRANK, at the end of this performance FRANK goes rouge and pulls of a stunt that involved calling everybody in the audience (more than 1000 people) and giving each of them a personalized message.

Anders was in charge of making this stunt work. It involved getting the list of participants with their phone numbers (on the day), formatting the list and removing any invalid information. A customized audio clip was then generated for each person greeting them by name and telling something related to the performance. These audio files were then uploaded to a server using FTP. In order to execute the phone calls Twilio's service was used. In order to use Twilio's API a XML file with instructions for where to find the audio clips needed to be generated for each person.

When it was go time, the “call script” had to make sure to follow the API’s restrictions, for example there was a limit on the amount of API calls per second, how many phone calls could be queued up and how fast they would be executed. All this was handled by a Python scripts created by Anders. The result was an amazing performance where you could hear the phones all over the venue started ringing. The stunt was repeated at several performances by Cecilie, with Anders doing the prep work and executing the script.

Anders took over the project from another team that had failed to complete the task, only a few days before the big event putting a lot of pressure on him. Keeping his head cold he could either try to make head or tail of the other team’s work, but not being familiar with the code Anders instead choose to build a solution from scratch. In the end his work saved the performance.

Keywords: Twilio API, Python, XML, FTP, Art Performance

Workshop facilitator - Hands on with oneWEX

IBM, Q2 2018 - Q2 2018

Arrow is IBM’s main distributor and often facilitates workshops and education on products from IBM. In connection with Watson Explorer (WEX) and its new platform oneWEX being able to understand Danish, Arrow desired to organize a hands on workshop with some of its business partners.

Anders and two colleagues were in charge of facilitating this workshop and designing some hands-on exercises that showcased what WEX is capable of. Due to the workshop being on short notice, oneWEX would have to be run locally since there wasn’t time to setup a proper server. A special developer version of oneWEX was deployed into a Docker image, so that it could easily be run of different machines in a Docker container.

Anders had the main responsibility of for this project, delegating the work between himself and his team, understanding the team dynamics and knowing each person’s strength and weaknesses.

Keywords: Docker, oneWEX, Workshop, Facilitation, Hands on

Full Stack Developer - Gesture Based Input Method for Wearable Devices

DTU, Q1 2018 - Q3 2018

Self-tracking is an important tool for the medical industry, when used to track patients psychological state (e.g. tracking experienced pain, mood or mental illness) it can give valuable insight that can be used to discover patterns not otherwise possible and be used to provide better treatment. The problem with self tracking is that it introduces a burden on the user. It is very important to minimize this burden.

Anders developed and tested a wearable device for self tracking using arm gestures to input specific values on a scale, this solution enables its user to self track only by adjusting the angle of their wrist and pressing a button on the device. An experiment in a controlled environment was conducted in order to compare the wearable device to a Visual Analogue Scale (VAS). The results showed that the wearable performed equally to the VAS.

On a technical level Anders first developed a proof of concept using a microcontroller, Bluetooth module and IMU. When it was shown that the concept worked a MetaMotionR Developer Kit by mbientlab was acquired. This device had the same capabilities as the PoC, but in small form factor. An Android App was developed that communicated with the MetaMotionR via Bluetooth. The app could import the data collected by the MetaMotionR, display the results and export it to .csv and send via email. Another more complicated app was created in order to conduct the experiment.

The results of the experiments were investigated using Python, and showed that the solution created by Anders performed equally to a VAS meaning that the device has great potential for real world use. The success of the project was due to Anders' wide technical knowledge enabling him to combine several different components, on top of this his knowledge of UX and HCI contributed a lot to the success of the project.

Keywords: HCI, UX, Android Studio, Arduino, Java, Python, mbientlab, Bluetooth, Android

Full Stack Developer - Remote Controlling House Appliances

DTU, Q1 2016 - Q2 2016

Internet of Things (IoT) opens the door for many possibilities, one of those being able to control and automate almost any aspect of one's home. The idea for this project was to create "Smart Blinds" that could be controlled from a smartphone and automatically roll the window blinds down and up.

Anders developed "Smart Blinds", an IoT device that can control the blinds based on time of day, brightness and commands from an app. The device was run on a Raspberry Pi and equipped with a motor to control the blinds and a photoresistor to sense the brightness. A companion app was developed in Android to control the device, settings timer for when the blinds should be up or down, setting a threshold for brightness and with the option to instantly roll the blinds up or down. The app and device communicated over the internet using either UDP or TCP. A comparison was made between the two protocols investigating power and data consumption.

Anders used his knowledge of network technology to design the protocols for the device and using his technical skillset he was able to successfully combine the different technologies to create a working "Smart Blind".

Keywords: IoT, Android Studio, Java, Python, Raspberry Pi, TCP, UDP

Full Stack Developer - Smart Lamp

DTU, Q2 2018 - Q2 2018

People often forget to turn of their lights, wasting energy. Having just one hue of white for a lamp is not ideal for all situations. The solutions is to developed a "Smart Lamp" that automatically will turn on and off and set the appropriate color temperature based on time of day, brightness and whether or not a person is present (based on movement).

A Raspberry Pi with sensors for brightness and movement controlled the lamp. It communicated with an Android app that could change the thresholds for brightness, movement and time of day to turn/off the lamp.

Anders was in charge of created this lamp, working on every aspect of it from design to implementation. His wide technical knowledge contributed to the success of this project.

Keywords: IoT, Android Studio, Java, Python, Raspberry Pi, TCP

UX Designer - Lean UX design process for "Smart Car Charger"

DTU & Tomorrow (tmrow.com), Q3 2017 - Q4 2017

DTU Electro and Tomorrow (who provide data on electricity's carbon footprint) are in collaboration in order to create a "Smart Charger" for electric vehicles. Once finished it will be able to charge a car in order to minimize it carbon footprint, based on what time it is most optimal to charge.

The problem is that the smart charger will be plug and play, and the user might not be informed of the benefit of using it. To prevent this, the design of the solution must get the message across.

In his role as UX designer, Anders was responsible for the iterative lean design process. First a consumer survey was conducted to get to know the user's thoughts on the topic. Then the process of creating mockups, A-B test and getting user feedback was conducted several times, before achieving a design proposal that satisfied both the user, DTU Electro and Tomorrow. Using his UX capabilities he was able to lead the project in the right direction.

Keywords: Lean UX, Iterative, A-B testing, Consumer surveys

Full Stack Developer - Multiplayer Ludo

DTU, Q1 2014 - Q1 2014

Recreating a digital version of the classic board game Ludo with multiplayer capabilities over the internet. The game was developed from the ground up, with a server in charge of the game and controlling both game logic and instant messaging. The server accepts up to 4 connections from clients. The clients display the GUI with the game board and instant messaging on the side, and updates the game board when receiving commands from the server.

Keywords: Java, Server, Client, Instant Messaging

Passion Projects

Location Tracking for Medic team at Smukfest festival

Anders has been a volunteer at the Smukfest festival for 2 years in a row. He worked together with a small team to develop a platform for tracking the location of the medics spread out over the festival. This was presented in a dashboard that the dispatch team can use to effectively assign the right medics closest to an accident, thus getting people the help sooner and potentially help safe their lives.

To achieve this, the location of the walkie talkies that the medics used where transmitted over a custom build LoRaWan network and stored in a database. This was then presented on a dashboard on top of a map view with custom location data that represent the festival area. All this ran in real-time providing up to date data for the dispatchers.

Exchange Rate Tool

When talking with his bank about restructuring his mortgage Anders learned that not only was the interest rate of the new mortgage important, but also the exchange rate of how much money goes into paying of the mortgage and how much is just consumed by the bank. Anders then discovered that while his bank publish this exchange rate on their website every 15 minutes, they don't offer any tool to keep an eye on it or show the history of the exchange rate, limiting people ability to chose a good time in which to restructure their mortgage. Anders not being they type to give in to the system decided develop his own tool. Using Flask (Python) he created a website that would scrape the exchange rate information from the bank, save it to his own personal storage and create a visual graph (using Bokeh) of the exchange rate over time to be displayed on the site. He also build in and alarm that would send him an email once the exchange rate surpassed a certain threshold. Anders then deployed all this first to AWS on Elastic Beanstalk, but later moved it to IBM Cloud to save money on hosting. The tool have given Anders the possibility to spot trends in the exchange rate and wait for the perfect time for restructuring his mortgage.

Cat of the Day - Slack Application

When joking around Anders and some colleagues thought it would be fun to have a tool that would provide you with a daily picture of a cute cat. Anders took this idea and made it a reality using a mixture of different services and technologies by scraping cats from Instagram, writing a backend that in a mix of PHP and Python hosted on both AWS and his personal webhotel that intergrated with Netlight's internal Slack workspace with a channel where the application would post its daily cat pictures. On top of this Anders also created an "instant cat" feature where any user could get a cat on demand posted anywhere using a custom Slack command. The application has been a huge success at Netlight where it is still running and provided people there daily cat fix.

Social Jeopardy, a Party Game

Anders enjoys playing games with his friends, combining his passion with his technical skillset he transformed a party game from pen and paper into the digital realm. The original game involve answering personal questions, which then are turned into answers in a Jeopardy style game. The digital solution solved three main flaws of the original game, players recognizing hand writing, players getting their own answers and unfairness of question distribution. This results in a much better and more fun game. The solution was created using Google forms to collect questions and the game was created in Python using TkInter.

Taking this to the next level Anders has rebuild the game from the ground up and deployed it to his own website, this involved using HTML, JavaScript and PHP with new challenges such as security needed to be handled.