Sprint period

The sprint period spanned weeks number 4, 5 and 6. More specifically starting the 26th of January and lasting until February 08th.

Goal

- Set up Figma and start design
- Read the project theme and discuss how we interpret it
- Create initial fronted application
  - Implement internationalization
- Create initial backend application
- Infrastructure
  - Install Kubernetes
  - Configure ArgoCD
  - Docker Hub

Work distribution

Daniel will be focused on setting up the server with Kubernetes and configuring ArgoCD since he has experience with it. Anders will learn how it works and how to use it. There will also be Docker hub repositories where docker images will be uploaded and later used in Kubernetes.

Anders will set up Figma and start creating the different pages, variables to enable the group to design effectively in Figma in the future. He will also setup the initial Next.js frontend application with Material UI. After setting up the application he will focus on implementing internationalization that will be used throughout the website.

Accomplishments

Under this sprint we accomplished setting up Figma with variables and different components that can be used to design all the pages in the future. To show this a hero section was created for the landing page. The group discussed and then agreed to design a professional looking website where search should be the main focus of the page. We decided that we wanted a modern top navigation, centred content that does not take up all horizontal space and a quite standard footer.

The project themes were revealed this sprint and the group discussed how to interpret it. The group got assigned the “Flights booking” theme. By looking at the project theme we decided that we wanted the main pages: landing, search and details. In addition to this we want a login page and a profile/settings page as well as a way for admins to administer the site.

We set up a simple Next.js application, set up themes, and made the application ready to develop components. We also set up internationalization so that we can serve different content for different languages. This was implemented in a way that the language is stored in the first “directory” of the URL (eg. /en/search). In addition to serve different content on the page, the internationalization can be configured to use different URLs for different languages as well as different metadata (eg. Title, description).

We installed Kubernetes from microk8s snap install. This is an install made for Ubuntu and fit our needs perfectly. After that ArgoCD was installed and configured. It has its own repository where configuration files are stored. GitHub Actions was setup to automatically build the application, Dockerize it with docker files and upload it to docker hub. This will be automatically synced to ArgoCD when changing the configuration files.

A simple backend application was made to test out the CI/CD pipeline and testing it on ArgoCD. But needs more work.