

# Metode

## Planlægningsværktøjer

I opgaven er der anvendt forskellige planlægningsværktøjer. Hjælpemidler som skal hjælpe med, at disponere min tid bedst muligt. Værktøjerne hjælper med, at danne et overblik over ens arbejde, og afklare hvor meget man mangler, at udarbejde samt hvor lang tid man har til de forskellige opgaver. Værktøjerne hjælper med, at give mig som studerende ro igennem processen.

Alt mit arbejde er samlet i google drev, så jeg kan arbejde på mine opgaver mens jeg er på farten. Yderligere har jeg lokalt haft GIT installeret som synkroniseres med indholdet fra google drev, dette har sikret mit arbejde med god backup fra starten, så intet kan gå tabt. Kombination af god struktur og grundig planlægning vil hjælpe med, at have ro igennem processen. Til at holde styr på mine kilder og sikrer mig god belæg for mine data. Har jeg fra starten anvendt zoteros web library til, at holde styr på mine kilder.

## Trello

Trello er et planlægningsværktøj hvor man kan oprette Boards for sine projekter. I hvert board har brugeren mulighed for, at oprette en liste over opgaver. Brugeren har herefter mulighed for at tilføje diverse egenskaber til opgaverne, som f.eks. "due date, checklister, kommentarer & labels". Det kan være en stor hjælp til, at styre opgaverne og holde deadlines for sit projekt, se bilag 1: Trello.

I opgaven har jeg anvendt Trello som planlægningsværktøj. I Trello har jeg skrevet en liste over de opgaver som skal udføres i forbindelse med rapport og produkt. Hver opgave har fået en due date, som hjælper mig med at dele arbejdet lige ud og disponere min tid bedst muligt. Nogle af punkterne er også blevet tildelt delopgaver, som opdelere større emner.

## Gantt kort

For at sikre mig min arbejdsplan og tidsplan fra trello er realistisk, og at tiden er disponeret ordentlig ud igennem hele forløbet har jeg udarbejdet et Gantt kort, se bilag 2: Gantt kortet indeholder alle arbejdsopgaver og markere hvornår jeg har afsat tid til den enkelte opgave. I Gantt kortet har jeg estimeret, hvor mange timer jeg regner med,

at bruge. Yderligere hjælper Gantt kortet mig med at overholde de deadlines jeg har afsat i Trello. Samt sikre, at der bliver afsat tid nok til alle opgaver.

## Udviklingsværktøjer

Applikation skal have et API som indeholder informationer omkring de transportmidler, der skal stå til rådighed ved Kolding lystbådehavn. Der er flere måder at stille data til rådighed online, et API er en af dem (API, 2019). Bruger skal kunne se om der er ledige transportmidler og have mulighed for “tage dem” hvis der er nogen ledige. Det vil fungere ligesom i et solcenter, hvor man træder ind af døren og kan vælge en ledig kabine, i dette tilfælde måske en cykel. Bruger skal derfor have mulighed for, at læse informationer fra et API, men også sende opdatering til API’et.

Som hjælpemidler til at programmere og sikre backup på projektet, anvender jeg GIT og github (SimonLervad/bachelor, 2020). Der findes forskellige måder, at anvende GIT på. Hvor jeg har valgt, at bruge Terminalen. Det har jeg fordi jeg i forvejen bruger Terminalen flittigt til, at køre applikationen og overvåge loggen for data output, requests og fejl. Jeg føler at det er her jeg har mest kontrol over de kommandoer der bliver udført, hvilket nogen gange kan være altafgørende, hvis man laver en fejl.

## Node.js Express

Jeg har valgt at bygge API’et i Node.js express som er et web framework der er minimalistisk og simpelt at arbejde med (Express A Web Framework, 2020), det er et framework som er nemt at gå til og bygge videre på (Express web framework (Node.js/ JavaScript), 2020). Jeg har valgt Node.js fordi det bygger på javascript og tillader javascript server side. Javascript er nemt at arbejde med og manipulere, men giver også en sammenhæng mellem frontend og backend. Begge dele kommer til, at spille stor rolle i den her applikation.

## Vue.js

For at inddrage ny viden fra praktik, har jeg valgt at bygge applikationen op i Vue.js. På npm har jeg fundet en udvidelse til Node.js Express som kører på Vue.js (express-vue, 2020). Jeg har valgt Vue.js fordi det giver god dynamik til programmøren og mulighed for at opbygge og genanvende komponenter. Vue.js er et framework bygget på javascript, så hele applikation vil blive opbygget i javascript. Hvilket giver god

gennemsigtighed i koden og ens struktur og opbygning i applikation hele vejen igennem. Vue.js sammen med node.js levere en onepage løsning som gør, at browseren ikke sender en ny request, når man navigere rundt på siden. På den måde sparer klienten tid på kommunikation med serveren, da siderne bliver renderet ved klienten. Den opbygger en virtuel DOM som ændrer sig i takt med at man navigere rundt, så brugeren for oplevelse af en alm. website. Vue.js sikrer, at et website opererer hurtigere.

## Vue storefront

Vue storefront er et PWA framework, som kan bruges til e commerce løsninger. Det består af et API som kan transportere data fra et CMS system som magento, woocommerce og shopify til en anden front end applikation. Når man anvender vue storefront, vil man have tre projekter. Det vil bestå af et CMS system, hvor alt data og database data er til gemt, både for produkter, brugere, emails, ordre og content som widgets, blocks og CMS sider. Så har man et vue storefront projekt, som indeholder frontend delen af ens løsning. Her imellem har man så Vue storefront API som er en bro for alt data fra backend i CMS til vue storefront front end.

### VUE STOREFRONT FEATURES

## Setting the new standard in eCommerce



### Open-source

With no license and vendor lock-in, the sky is the limit: build what you like and you'll always own your site.



### Headless and API-focused

Connects to any backend and allows you to merge with any third-party tools.



### Progressive Web App

Provides shopping experiences like native apps while boosting site performance and customization

(Headless CMS | Vue Storefront, 2020)

Hvis løsningen skal udarbejdes i magento CMS, så vil vue storefront være et stærkt valg.  
Har arbejdet med begge løsninger i min praktikperiode hos Improving ApS.

## MongoDB & Mongoose

Der findes mange typer databaser, både MySQL og NoSQL. Samt forskellige måder du kan administrerer dine databaser på. Når det kommer til Node.js Express er MongoDB det

oplagte valg. MongoDB giver mulighed for at oprette kollektioner bestående af JSON data, typisk objekter.

Da JSON minder meget om javascript og nemt kan kombineres med javascript applikationer. Arbejder vi igen i samme stil, som opretholder god struktur og opbygning igennem hele applikationen. Med mongoose kan vi bygge skemaer som definere indholdet af de objekter vi gerne vil have i databasen. Som sikrer, at det kommer i rigtig format og at variablen altid holder samme datatype, dette giver mindre fleksibilitet, men bedre sikkerhed og stabilitet. Som udvikler ved vi aldrig om brugeren forstår hensigten, vi må derfor ikke give dem mulighed for at lave fejl. Hvor defineret datatyper samt validering af input sikrer, at dette ikke sker.

## Magento

Magento er et professionelt ecommerce framework, som primært bliver anvendt til webshops. Magento er kendt som et content management system ligesom woocommerce og shopify (Magento Commerce, 2020). Løsningen kan levere en platform som er nem at integrere med og som tilbyder udviklere, at lave deres egen løsning. Det er global kendt og kan styre shops som kan køre i hele verdenen. Jeg har igennem min praktikperiode hos Improving ApS arbejdet med Magento webshops. En webshop kunne være en løsning til Kolding lystbådehavn, hvor man ”sælger” et transportmiddel, i x antal timer. Løsningen kan installeres på deres nuværende domæne, under et subdomæne. De kan opretholde deres nuværende løsning, men samtidig implementere en ny løsning til noget udlejning.