

# Cupcake

## Gruppe 14

### Devs

Navn	Email	Github
Esben Oliver Moldt	cph-em170@cphbusiness.dk	Esben10
Carl Emil Styrbjörn Gullacksen	cph-cg207@cphbusiness.dk	Styrse

### Klasse:

I-dat-da 0924b - Sem2 - (Klasse B)

Udarbejdet: Marts/april 2025

### Projekt:

Github repo: <https://github.com/Styrse/Cupcake>

YouTube gennemgang: <https://youtu.be/3qcgoDqoYR4>

## Indholdsfortegnelse

<b>Indledning:</b>	<b>2</b>
Vores kunde:	2
Olskers krav:	2
Teknologivalg:	3
Projekt konfiguration	3
Backend	3
Frontend	3
Database	3
Testing	3
<b>Vision:</b>	<b>4</b>
Udvidelse	4
<b>User Stories</b>	<b>5</b>
Tidsestimat	5
User Stories (Minimum)	5
User Stories (Extra)	6
Planlægning	6
Rapportering	6
Handover & Implementering	6
Samlet tidsoversigt	7
<b>Mockup design:</b>	<b>7</b>
<b>ER-diagram:</b>	<b>8</b>
<b>ERD Cupcake</b>	<b>9</b>
User tabel	9
Order tabel	9
Product tabel	9
Cupcake tabel	9

Cupcake_bottom, Cupcake_top tabel.....	9
<b>NavigationsDiagram:.....</b>	<b>10</b>
<b>Særlige forhold.....</b>	<b>10</b>
Session.....	10
Validering af input.....	10
<b>Status på implementation.....</b>	<b>11</b>
Ikke implementerede ekstra features tidsestimat.....	11
Implementerede ekstra features.....	12
Kendte bugs.....	13
<b>Anerkendelser og Referencer.....</b>	<b>13</b>
<b>Process.....</b>	<b>13</b>

## Indledning:

Cupcake projektet har til formål at tilgodese en kundes ønske om at få en funktionel website der kan bruges til at bestille cupcakes. Der skal både oprettes front-end og back-end som understøtter denne webservice så vi opstiller både en database til at gemme alle nyttige oplysninger om kunder og ordre til klienter, og så udvikler vi en flot og overskuelig brugerflade så kunder i diverse målgrupper kan finde ud af at bruge servicen.

Der bliver også lavet et fint dashboard til Employees og Admins til brug i analyse samt håndtering af ordre og kunder

## Vores kunde:

Virksomheden som har bestilt dette website er Olsker Cupcakes. De er et økologisk bageri fra Bornholm som har et ønske om at udvide deres sortiment.

De vil gerne have en simpel web service til deres firma, som gør det muligt at bestille specifikke cupcakes. Deres brugere skal have mulighed for at vælge mellem en række cupcake-bunde og cupcake-toppe for at bestille præcis den variant der ønskes.

## Olskers krav:

- Der skal som kunde være mulighed for at oprette en profil.
- Kunde- og ordreoplysninger skal gemmes i en database.
- Når en kunde har lagt bestilling, skal den gemmes på denne kundes egen profil og de vil dernæst have mulighed for at gå fysisk ned i butikken og hente deres bestilling.
- Der skal være en administrator login som har adgang til informationerne i databasen såsom alle kunders ordrer så de kan redigere i disse alt efter nødvendighed.
- Kunder skal også selv kunne logge ind og redigere deres egne ordrer efter behov.

## Teknologivalg:

### Projekt konfiguration

Projekt konfiguration	Version
sourceEncoding	UTF-8

### Backend

Backend	Version
Java	17
Javelin	6.5.0
ThymeLeaf	3.1.3
ThymeLeaf extras	3.0.4
sl4j	2.0.17
jackson	2.18.3

### Frontend

Frontend	Version
HTML	Nyeste version
CSS	Nyeste version
JavaScript	Nyeste version

### Database

Database	Version
Postgressql	42.7.5
Docker	Nyeste version
hikariCP	6.2.1

### Testing

Testing	Version
JUnit	5.12.0
hamcrest	3.0

# Vision

Visionen med programmet er at skabe et nemt og lækkert user interface (UI) for kunder og ansatte der skal bruge systemet. Vi har blandt andet haft fokus på at programmet skal være lækkert, intuitivt og nemt at bruge.

Olsker vil med systemet kunne forstå deres kunders købe mønstre bedre. Dette vil de så kunne bruge til at analysere fx hvilke dage der er travlere, så der kan bages ekstra cupcakes for at undgå at løbe tør i løbet.

Ejerne af Olsker Cupcakes vil også kunne følge hvilke smage der er populære og hvilke der måske ikke sælger så godt og derfor måske skal tages af menuen.

Der vil også på sigt være mulighed for at udvide med nye produktgrupper som behovet opstår. Det kan være andet spiseligt eller fx drikke.

## Udvidelse

Programmet kan udvides i etaper med forskellige features der alle vil kunne forbedre og automatisere mange arbejdsopgaver for Olsker Cupcakes. Nogle af disse, som ofte er tidskrævende, vil være en integration til et regnskabsprogram, der automatisk overfører salg, hvilket vil gøre regnskabet hurtigere og mindre tidskrævende. Dette vil kunne frigøre tid for ejerne til det der er kernen i deres forretning.

Der vil være mange udvidelser der vil give stor værdi til forståelsen af den økonomiske side af Olsker Cupcakes. Disse vil kunne blive tilføjet hyppigt med stor værdi for at kunne forstå forretningen bedre.

Under punktet "Ikke implementerede ekstra features" er der en lang liste af punkter der kan tages udgangspunkt i. Listen har to prioriteter. Den ene tager udgangspunkt i et finansielt perspektiv alt efter hvor finansielt vigtigt det vil være og det andet er for brugeroplevelsen. Opgaverne er prioriteret fra 1 til 5, hvor 1 er det vigtigste og 5 det mindst vigtige. Der er også tilføjet et tidsestimat for hver opgave.

# User Stories

**US-1:** Som kunde kan jeg bestille og betale cupcakes med en valgfri bund og top, sådan at jeg senere kan køre forbi butikken i Olsker og hente min ordre.”

**US-2** Som kunde kan jeg oprette en konto/profil for at kunne betale og gemme en ordre.

**US-3:** Som administrator kan jeg indsætte beløb på en kundes konto direkte i Postgres, så en kunde kan betale for sine ordrer.

**US-4:** Som kunde kan jeg se mine valgte ordrelinjer i en indkøbskurv, så jeg kan se den samlede pris.

**US-5:** Som kunde eller administrator kan jeg logge på systemet med email og kodeord. Når jeg er logget på, skal jeg kunne se min email på hver side (evt. i topmenuen, som vist på mockup'en).

**US-6:** Som administrator kan jeg se alle ordrer i systemet, så jeg kan se hvad der er blevet bestilt.

**US-7:** Som administrator kan jeg se alle kunder i systemet og deres ordrer, sådan at jeg kan følge op på ordrer og holde styr på mine kunder.

**US-8:** Som kunde kan jeg fjerne en ordrelinie fra min indkøbskurv, så jeg kan justere min ordre.

**US-9:** Som administrator kan jeg fjerne en ordre, så systemet ikke kommer til at indeholde udgyldige ordrer. F.eks. hvis kunden aldrig har betalt.

## Tidsestimat

### User Stories (Minimum)

User Stories (Minimum)	Tid	Prioritet
US-1	8-12	1
US-2	6-8	5
US-3	4-6	6
US-4	6-8	2
US-5	6-8	3
US-6	4-6	4

Total tid: 34-48 timer

## User Stories (Extra)

User Stories (Extra)	Tid (Timer)	Prioritet
US-7	4-6	2
US-8	3-5	1
US-9	3-5	3

Total tid: 10-16 timer

## Planlægning

Planlægning	Tid(Timer)
Kravanalyse	2-4
Systemarkitektur & Design	3-5
Opdeling af opgaver	2-3

Total tid: 7-12 timer

## Rapportering

Rapportering	Tid(Timer)
Statusopdatering	4-6
Test & fejlsøgning	4-6
Afsluttende rapport	4-6

Total tid: 12-18 timer

## Handover & Implementering

Handover & Implementering	Tid(Timer)
Onboarding & Dokumentation	3-5
Workshop/træning	2-3
Support	1-2

Total tid: 6-10 timer

## Samlet tidsoversigt

Fase	Tid(Timer)
Udvikling(US-1 til US-6)	34-48
Planlægning, Rapportering & Handover	25-40

Total tid: 59-88 timer

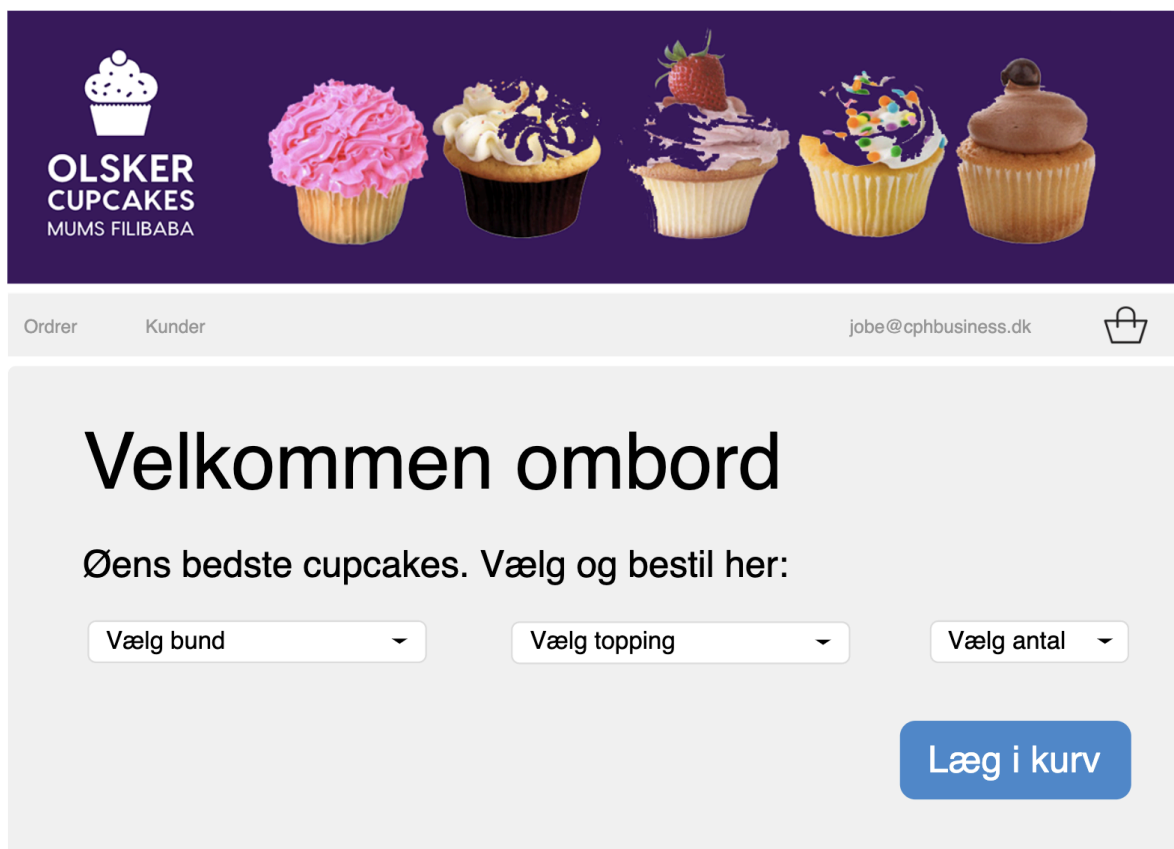
Total tid: 4-6 dage

Fase	Tid(Timer)
Udvikling(US-1 til US-9)	44-64
Planlægning, Rapportering & Handover	25-40

Total tid: 69-104 timer

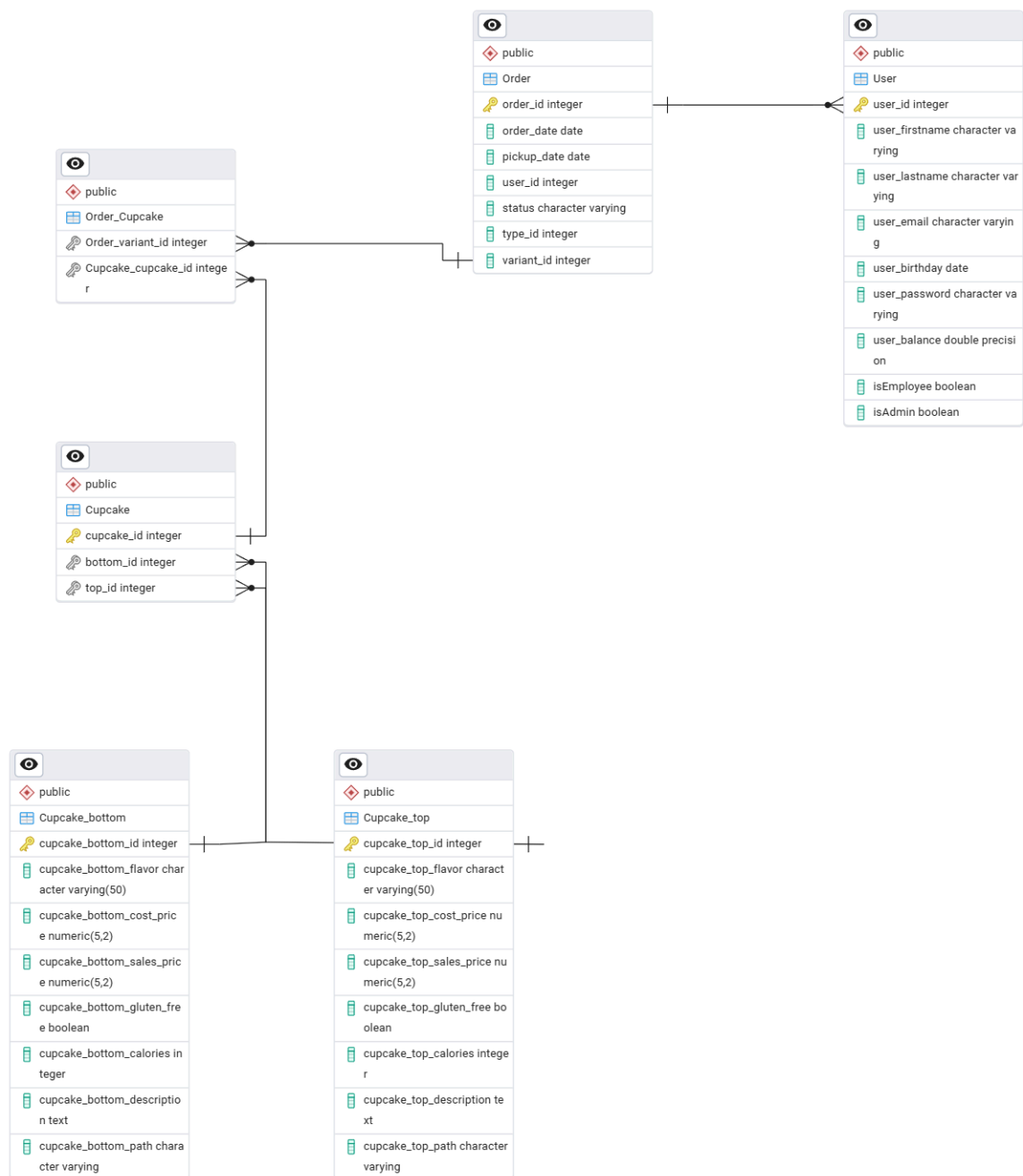
Total tid: 6-9 dage

## Mockup design:



Vi vil tage udgangspunkt i dette mockup design fra kunden med tydelig funktionalitet af websitet og simple UI.

ER-diagram:





# ERD Cupcake

## User tabel

Vi har en User tabel der er opsat til at kunne indeholde de mest basale informationer om en User(email, password, firstname, balance). Så har vi også tilføjet to kolonner med "isEmployee" og "isAdmin" for at kunne tjekke hvilken type User vi har med at gøre så vi i Java kan lave objekter af de korrekte instanser. Derudover har vi tilføjet flere kolonner for senere at kunne implementere flere features som tiden/budgettet tillod det(lastname, birthday).

## Order tabel

Tabellen er der hvor vi forbinder User tabellen og Product tabellen samt holder styr på hvilken payment\_type og order\_status en ordre har.

## Product tabel

Denne tabel forbinder Order tabellen og Cupcake tabellen samt holder styr på hvor mange af hver Cupcake der er til en ordre.

## Cupcake tabel

Denne tabel består af et product\_id der forbinder til Product tabellen. De to andre er id's i henholdsvis Cupcake\_bottom og Cupcake\_top.

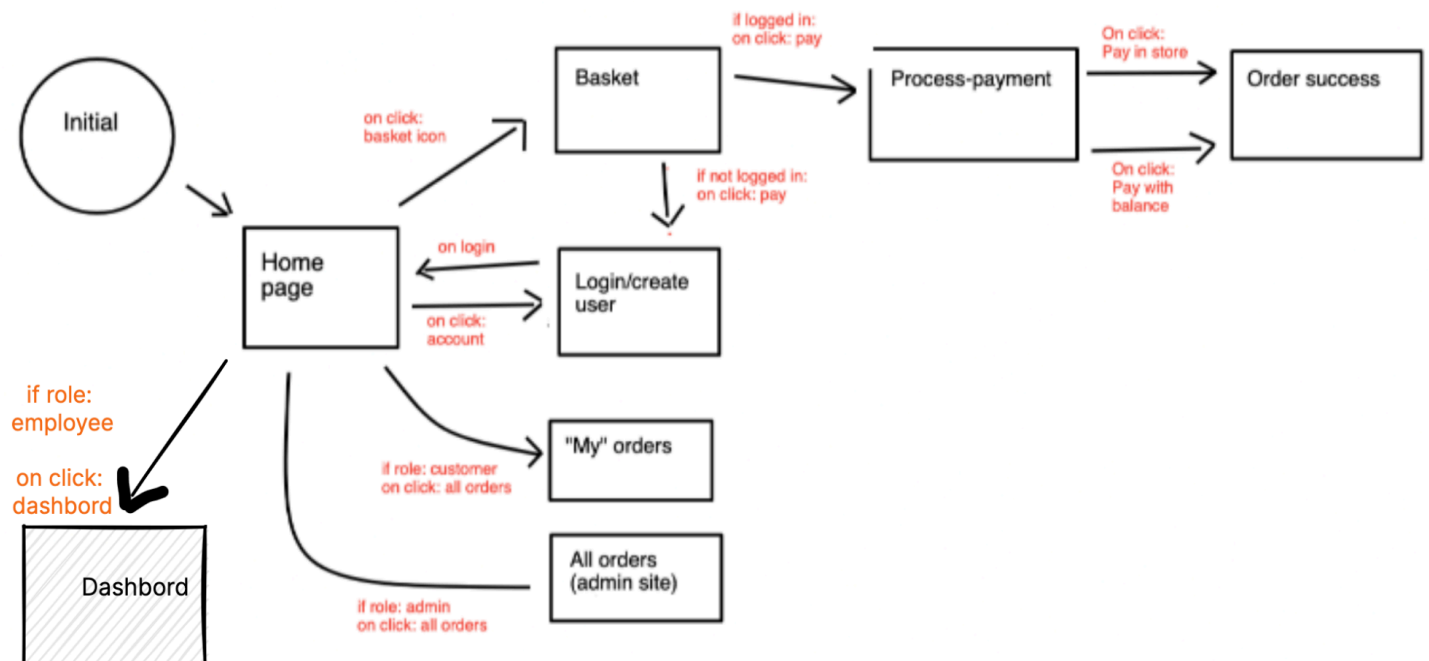
## Cupcake\_bottom, Cupcake\_top tabel

Disse to tabeller indeholder de samme kolonner og derfor beskrives de sammen. De informationer der som minimum er brug for er "id", "flavor", "sales\_price". Id bliver auto genereret i databasen. De tre andre parametre er blevet givet af kunden.

Derudover er der også lavet følgende kolonner "cost\_price", "gluten\_free", "calories" samt "description" der kan bruges senere til ekstra features alt efter tid og budget.

Der er også tilføjet en kolonne med en path til et billede der bliver brugt når der kan vælges en smag på hjemmesiden.

# NavigationsDiagram:



## Særlige forhold

### Session

User og Basket bliver gemt i session.

### Validering af input

Hvis en bruger opretter en ny bruger eller ændre deres email tjekkes der for om denne findes i systemet i forvejen.

# Status på implementation

## Hovedkrav

US-1	Kunden kan bestille og betale for cupcakes med valgfri bund og top.
US-2	Kunden kan oprette en konto og kan betale og gemme en ordre.
US-3	Admin kan indsætte beløb på kunders konto (balance)
US-4	Kunden kan se valgte ordrelinjer i en kurv med en samlet pris.
US-5	Brugere kan logge på med email og kode. Email er vist i hjørnet.
US-6	Admin kan se alle ordrer i systemet
US-7	Admin kan se alle kunder og alle kunders ordrer.
US-8	Kunden kan fjerne en ordrelinje fra sin kurv og justere sin ordre.
US-9	Admin kan fjerne en ordre i tilfælde hvor det er nødvendigt.

Alle de umiddelbare funktionelle krav er hermed implementeret. Der er få mangler af ekstra user-stories, da de er nedprioriteret for de 9 hovedkrav. Disse funktioner er der blevet gjort plads til at implementere senere, skulle kunden ønske en udvidelse.

## Ikke implementerede ekstra features tidsestimat

User Stories	Tid(Timer)	Finansiell prioritet	UX prioritet
Afhentningsdato	6-8	1	2
Leveringsdato	6-8	1	2
Dashboard	12-16	2	4
Salgs forudsigelse	12-16	2	3
Redigering af ordre	6-8	3	2
Download kvittering	3-5	4	1
Gensend kvittering	3-5	4	1
Medarbejder ordrestatus håndtering	4-6	2	4
Tilføj nye bunde/toppe	12-16	2	4
SuperAdmin funktioner	6-8	4	4
Password sikkerhed	5-7	5	5
Rabatkupon	5-7	2	4

Gavekort	5-7	2	3
Instagram feed	7-9	3	3
Ordre kommentar	3-5	4	2
Udvidet information	8-10	4	2
POS	20-25	1	5
Sprog	10-12	4	4
Meddelelser	5-7	4	2
Ordrebekræftelse	2-4	3	1
Opdateringsmail	2-4	3	1
Kvitteringsmail	2-4	3	1
Implementering af nye varegrupper	6-8	1	3

## Implementerede ekstra features

1. Betaling
  - MobilePay
  - Cash
2. Profil
  - Brugerprofil
3. Item
  - Varer har fået ekstra attributter der er gjort klar til senere brug
4. Account types
  - Employee klasse
  - Admin klasse
5. Footer
  - Kontaktinformation
  - Trustpilot forbindelse
  - Google maps
6. Ordre
  - Ordre status
7. Slet kunde
  - Employees kan slette kunder
8. Sider
  - Nogle sider er "sikrede/gemte" hvis en bruger ikke er logget ind
9. Log ud funktion
10. Brugerprofil opdatering
  - Ændre email
  - Ændre kodeord
11. Visuel cupcake bygger

## 12. Udvidede bruger begrænsning

- Nogle sites er ikke tilgængelige for brugere alt efter deres adgangsniveau

## Kendte bugs

### 1. Firstname

Firstname bliver ikke vist når hele User listen bliver vist for Employees.  
Der er en relation mellem dette og brugere der er en instans af Employee

### 2. Hover

- Nogle dele er ikke optimeret til medier hvor man ikke kan hover

## Anerkendelser og Referencer

Der er en tekstfil ("contributors") i mappen rapport der henviser til en liste af materiale der er blevet brugt til projektet. Filen indeholder referencer til ikoner, billeder, kode og andre ressourcer der er blevet brugt i forbindelse med projektet.

## Process

Da vi kun har været to i gruppen der har arbejdet godt sammen, har vi fokuseret på at kode og mindre tid på mock up i starten. Vi har stadig planlagt setup af website, tegnet og diskuteret hvordan vi kom i mål med MVP og hvilke features vi kunne nå at få med. Da vi kun var to og vi begge var helt enige i målet og vejen dertil endte rollerne med at blive lidt udvisket hos os i de to uger.

Undervejs programmerede vi sammen, derudover havde vi hver især opgaver vi arbejdede individuelt på.

Til fremtidige projekter vil vi blive bedre til løbende at dokumentere hvilke overvejelser og fremskridt vi har gjort for at sikre os at vores rapport bliver så præcis og fyldestgørende som muligt.