

# Olsker Cupcakes

En webshop

af

Peter Lønquist Thomasen

[cph-pt123@cphbusiness.dk](mailto:cph-pt123@cphbusiness.dk)

Thomas Schwencke Overgaard

[cph-to83@cphbusiness.dk](mailto:cph-to83@cphbusiness.dk)

-

Flow 4: Hold C, Gruppe 8

I perioden

19/04/2021 - 28/04/2021

Github:

github@Schwencke

github@PeterDaniel3D

<https://github.com/Schwencke/Cupcake>

Droplet:

(Peter) <http://46.101.212.228:8080/CupCake-0.1/>

(Thomas) <http://157.230.20.13:8080/CupCake-0.1/>

## Indholdsfortegnelse

Indledning.....	3
Baggrund.....	3
Teknologi valg.....	4
Krav.....	5
Aktivitetsdiagram.....	6
Domæne model .....	7
ER diagram.....	8
Navigationsdiagram .....	9
Særlige forhold .....	10
Sessionslag.....	10
Applikationslag .....	11
Andet .....	11
Status på implementering.....	12
Proces .....	12

## Indledning

Denne rapport beskriver udarbejdelsen af Olsker Cupcakes nye webshop. Olsker Cupcakes er en fiktiv virksomhed, der producerer egne hjemmelavede lækre cupcakes, som sælges fra butikken i Olsker.

Vi antager, at bestyrelsen efter nøje overvejelser i virksomheden har taget beslutning om at få lavet en moderne, men klassisk webshop, så butikkens kunder kan bestille cupcakes til afhentning i Olsker.

## Baggrund

Virksomheden Olsker Cupcakes sælger til daglig cupcakes fra deres forretning i Olsker. De har et ønske om at få en webshop, så deres kunder på forhånd kan bestille og betale cupcakes til afhentning.

Olsker Cupcakes ønsker at man kan sammensætte cupcakes med bund og topping, samt bestemme antallet af denne. Den valgte cupcake overføres til en kurv, hvorfra kunden kan tilføje flere cupcakes eller fortsætte videre til betaling. For at kunne betale for sin bestilling kræves det at man er oprettet som bruger.

Forretnings ansatte skal ligeledes kunne oprettes som brugere, så de kan få adgang til relevante kundeoplysninger og bestillinger.

Kravspecifikationer til webshoppens funktionalitet er udspecificeret i afsnittet om krav.

## Teknologi valg

Olsker Cupcakes er en demonstration af en JavaEE webapplikation, udviklet til at køre på en Tomcat webcontainer vha. Java, MySql, HTML, CSS, Twitter Bootstrap.

Webshoppen bygger på denne startkode: <https://github.com/ionbertelsen/sem2-startcode>

<b>IntelliJ</b>	<b>2021.1</b>
<b>JAVA</b>	<b>8</b>
<b>Maven</b>	<b>1.8</b>
<b>JDBC MySQL connector</b>	<b>8.0.19</b>
<b>Bootstrap</b>	<b>4.3.1</b>
<b>Junit</b>	<b>5.7.1</b>
<b>Hamcrest</b>	<b>1.3</b>
<b>JavaEE api</b>	<b>7.0</b>
<b>JSTL</b>	<b>1.2</b>

## Krav

Kunden har følgende kravspecifikationer.

### **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 MySQL, så en kunde kan betale for sine ordrer.*

### **US-4:**

*Som kunde kan jeg se mine valgte ordrelinier 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 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.*

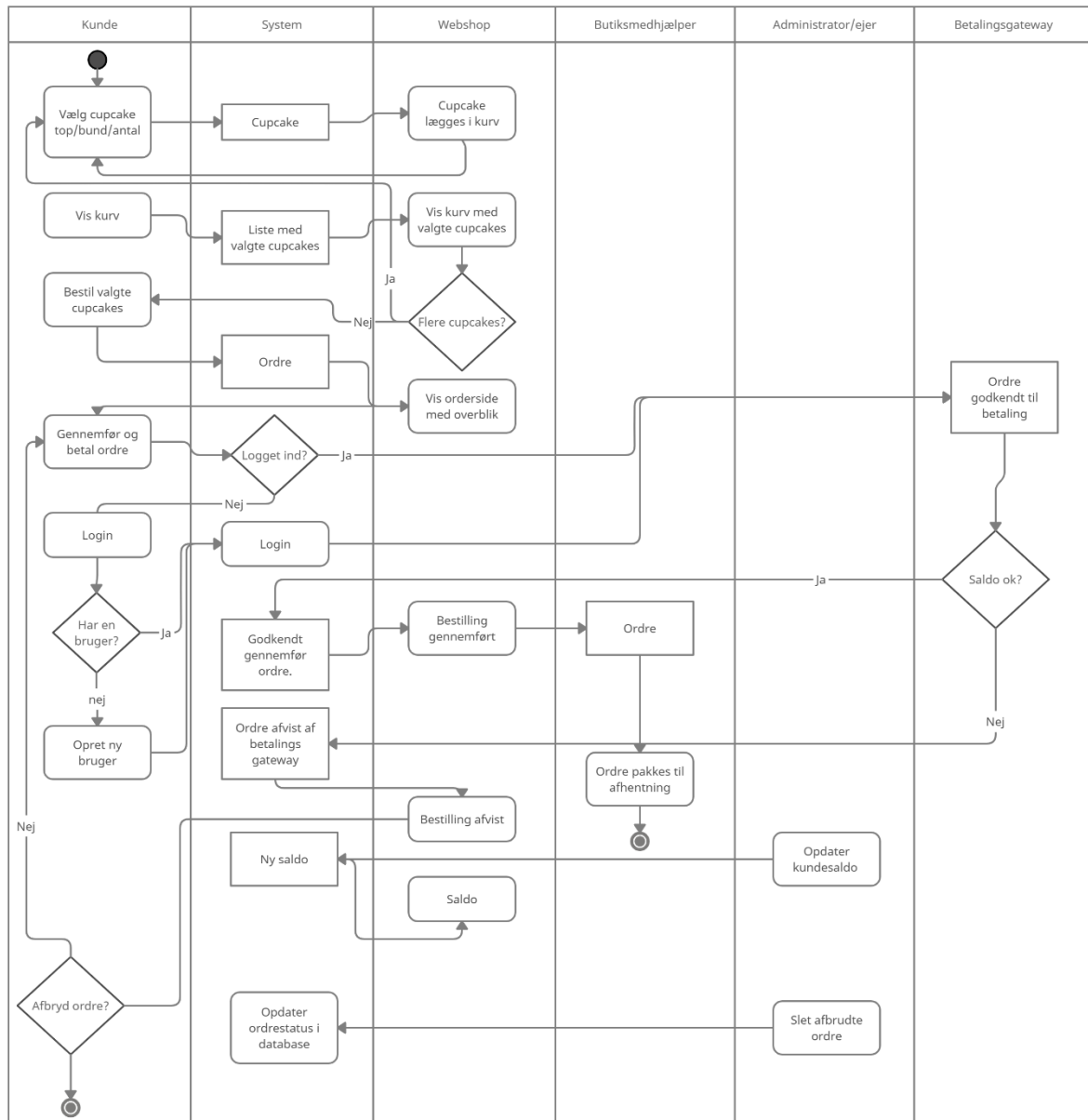
### **US-8:**

*Som kunde kan jeg fjerne en ordre fra min indkøbskurv, så jeg kan justere min ordre.*

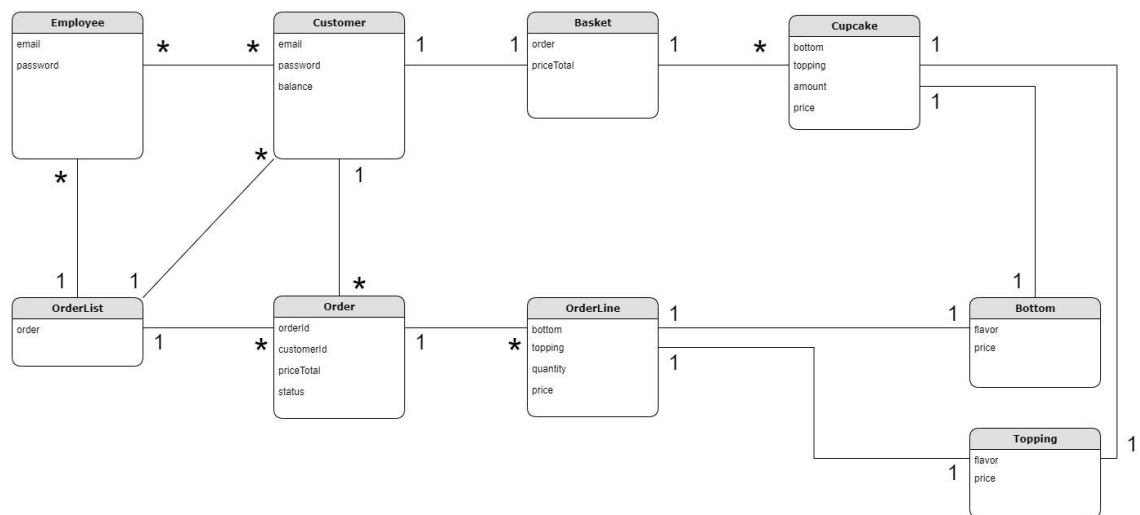
### **US-9:**

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

# Aktivitetsdiagram



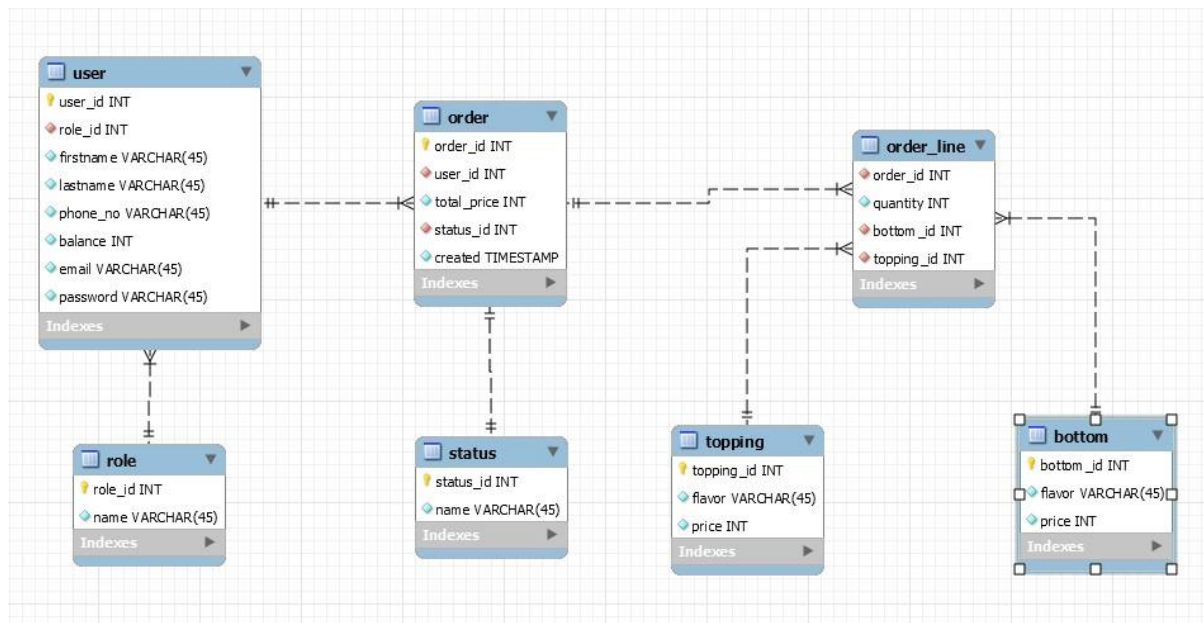
## Domæne model



Domænemodellen er lavet med udgangspunkt i kravspecifikationerne.

Der er blevet brugt en del tid på at få udarbejdet og diskuteret sammensætningen af klasserne i relation imellem Basket, Cupcake, Bottom, Topping og Orderline. Og i forhold til hvordan vi har skulle opsætte databasen.

## ER diagram

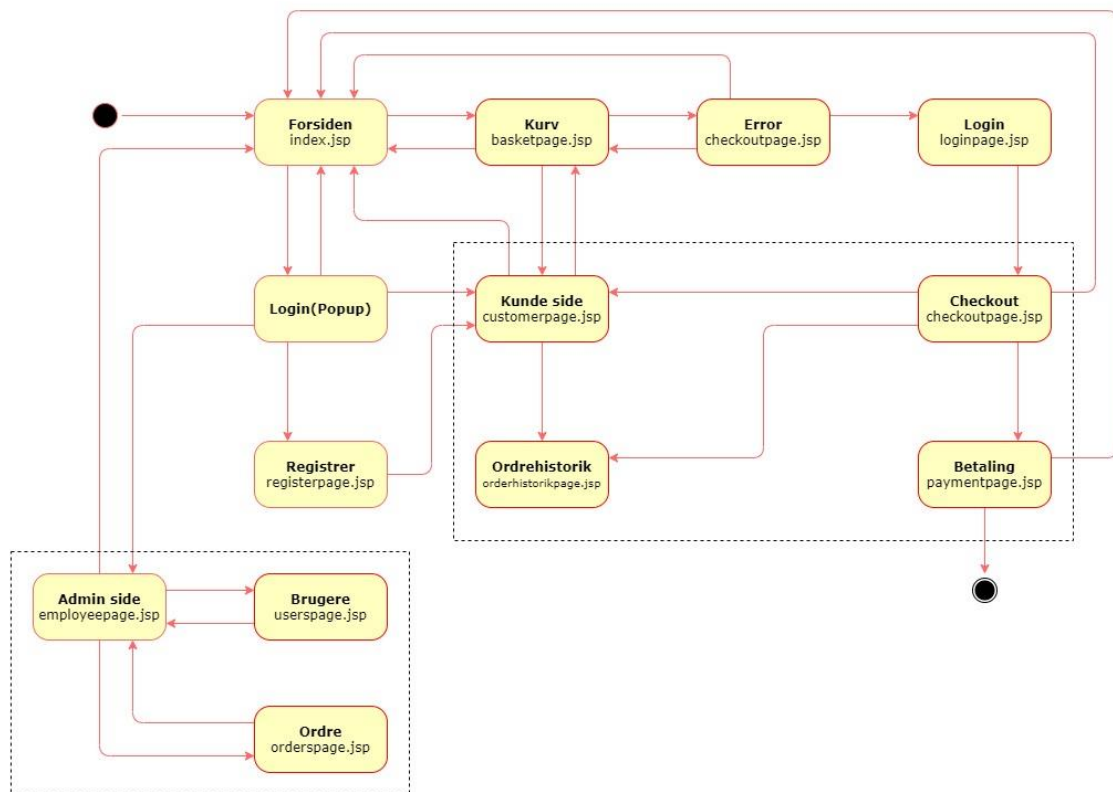


Databasen er sat op i 3. normalform.

Primærnøglerne på role, status, topping, bottom er ikke auto-genereret nøgler, da det antages at kunden allerede har et POS-system (Point-of-Sale), hvor disse værdier indgår.



## Navigationsdiagram



## Særlige forhold

### Sessionslag

I sessionslaget gemmes der følgende:

- **Orderlinelist** – Her gemmes der bund, top og antal af den sammensatte cupcake.  
*Benyttes af kurven til efterfølgende af samle en ordre.*
- **Pricetotal** – Her gemmes den samlede pris.  
*Bruges til at indeholde udregning for én enkelt cupcake, eller total på baggrund af antal af orderlines.*
- **Orderlinecount** – Viser antallet af orderlines i kurven.  
*Bruges til at vise antallet af cupcakes i parentes ved siden af Kurv i navigationsbaren.*
- **Orderlist** – Indeholder list af alle ordre.  
*Bruges til at liste alle ordre for enten administrator eller den enkelte bruger.*
- **Userlist** – Indeholder liste af alle oprettede brugere.  
*Bruges til at liste alle brugere for administrator. Benyttes til at opdatere saldo og evt. andre brugeroplysninger (Sidstnævnte er ikke implementeret).*
- **User** – Indeholder brugeren, hvis brugeren er logget ind.  
*Benyttes af webapplikationen til at håndtere bestillinger.*
- **Role** – Hvilken rolle har brugeren.  
*Benyttes udelukkende til at login.*
- **Email** – Hvilken email er brugt til login.  
*Benyttes til at login, samt visning i navigationsbare*

## Applikationslag

Applikationslaget bruges til at holde globale data hentet fra databasen.

Følgende initialiseres ved opstart:

- **Bottomlist** – Indeholder smag og pris.  
*Bruges til at liste indhold i rullepanel på bestillingssiden.*
- **Toppinglist** – Indeholder smag og pris.  
*Bruges til at liste indhold i rullepanel på bestillingssiden.*
- **Statuslist** – Indeholder navne på forskellige ordrestatus og id.  
*Bruges til at vise navnet på status ud fra id på en ordre.*
- **Rolelist** – Indeholder navne på forskellige roller og id.  
*Bruges til at vise navnet på rollen ud fra id fra en bruger.*

## Andet

På nuværende tidspunkt er der ikke valgt at lave validering af brugerinput, sikkerhed på databasen og håndtering af exceptions.

## Status på implementering

Vi kommet godt i mål med funktionaliteten af webshoppen ud fra givne use cases. Overordnet set er sitet stylet, dog med mangler på enkelte sider.

Selvom alle kravspecifikationerne er blevet implementeret og hjemmesiden er testet, så kan der forefindes fejl som vi ikke har haft tid til at rette. Man kan bl.a. bestille en tom ordre når man er logget ind og indtaster url til betalingssiden.

## Proces

Det blev besluttet at arbejdsformen for projektet skulle foregå i digitalt fællesskab 9 – 15, mandag til fredag. Vi har hver dag mødt ind på Zoom, og har på skift stået for at kode. Projektet er 100% udarbejdet i fællesskab, og koden er velovervejet og blevet vendt og drejet ud fra fornuftig diskussion. Det har hjulpet os igennem ved nye opståede problemstillinger, hurtigt og (forholdsvis) smertefrit. Vi har løbende forventningsafstemt med hinanden.

Den første dag blev brugt til at komme helt i mål med det praktiske; Domænemodel og database. På den måde kunne vi holde et godt overblik ved virkeliggørelsen af usecases inden for projektets tidsramme.

Kl. 14.59 fredag eftermiddag var sidste usecase implementeret og testet. Ved eftertanke er vi lidt imponerede over, at vi kunne overholde vores egen deadline med den tid vi også valgte at prioritere til diskussioner og overvejelser undervejs.

Alt i alt synes vi forløbet er gået rigtig godt. Vi mener begge, at denne proces har styrket vores samarbejde. Nuværende, men også fremtidige.