

Olsker Cupcakes

Navne:

Ermin Dzafic
Daniel Pedersen
Jens Christian

Email:

Daniel: cph-dp134@cphbusiness.dk
Ermin: cph-ed79 @cphbusiness.dk
Jens: cph-jo163@cphbusiness.dk

Github:

Incognito95
ErminDZ
Mawl026

Klasse: B

Tidspunkt:

Mandag 26. April 2021

Indledning:	3
Baggrund:	3
Teknologi valg:	3
Krav:	4
User stories:	4
Aktivitetsdiagram:	5
AS-IS:	5
TO-BE:	6
Domænemodel og ER Diagram:	6
Domænemodel:	6
ER Diagram:	7
Navigation Diagram:	7
Særlige forhold:	8
Status på implementation:	9
Process:	9

Indledning:

Projektet handler om at man skal kunne bestille cupcakes online så kunderne kan hente det fysisk i butikken. Vi blev uddelt noget startkode til at få noget at arbejde ud fra, vi videreudviklede systemet ved at tilføje de forskellige funktionalitetskrav, hvor kunder kunne oprette deres egen profil, ændre brugernavn og kodeord, bestille cupcakes, se deres ordrer og gennemføre bestil og betaling.

Baggrund:

Den type virksomhed som skal bruge systemet er et bageri i Bornholm og København. Til at kunne bruge systemet skal en kunde bestille og betale cupcakes med en valgfri bund og top, sådan at jeg senere kan køre forbi butikken i Olsker og hente sine ordre. Man skal også kunne oprette en profil for at kunne betale og gemme sine ordre, som kunde kan man logge på og se sin email i toppen af menuen såvel som fjerne en ordre fra sin indkøbskurv så man kan justere sine ordrer.

Teknologi valg:

- IntelliJ IDEA 2020.3.3 (Ultimate Edition)
- MySQL Workbench 8.0 CE
- Apache Tomcat 9.0.44
- Adobe XD 39.0.12.12
- HTML5
- CSS3
- Github/Github Desktop - 2.7.2
- Git Bash/terminal
- Bootstrap 5
- Digital Ocean - Droplet

Krav:

Hvad er firmaets håb med dette system (hvad er deres vision for systemet eller hvilken værdi er det jeres system skal tilføre deres virksomhed)?

Firmaets håb med dette system er, at gøre det nemmere for kunder, at bestille Cupcakes samtidigt med, at det er mere overskueligt for ansatte, at følge med bestillinger og kunder.

Der er også forventninger om systemet skal føre til øget salg, styrkning af virksomhedens brand, større kunde eksponering mm.

Visionen med systemet for virksomheden er, at tilføje mere funktionalitet på hjemmesiden samt, at kunne trække eventuelle statistikker fra databasen for, at bedømme hvilke Cupcakes er det mest sælgende og bruge denne information til, at forbedre produkterne og udvalget i sortimentet for, at øge salg i virksomheden.

Løbende opdateringer og forbedringer til hjemmesidens layout, design og funktioner, samt gøre det muligt, at kunne udvide hjemmesiden til nye ideér til udvidelse af firmaets produkter, til at indeholde mere end kun cupcakes.

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 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.

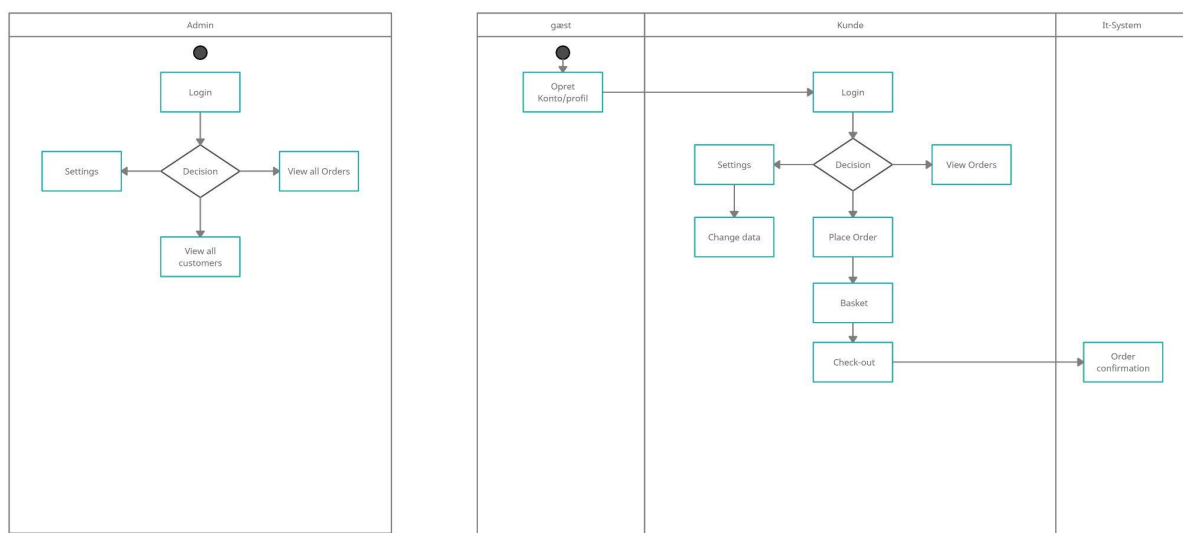
US-7: Som administrator kan jeg se alle kunder i systemet og deres ordrer, sådan at jeg kan følge op på ordrer og holde styr på mine kunder.

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 udgyldige ordrer. F.eks. hvis kunden aldrig har betalt.

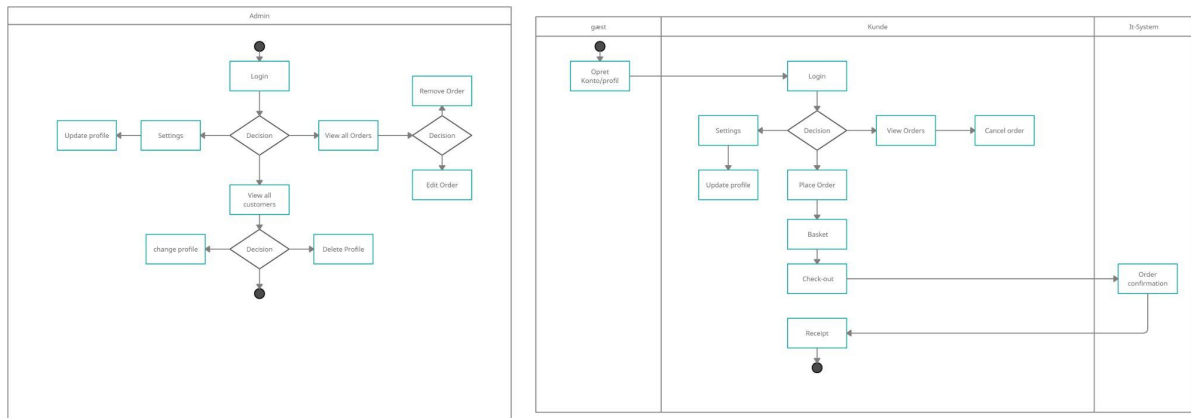
Aktivitetsdiagram:

AS-IS:



Aktivitets diagrammet heroppe viser hvordan vores AS-IS diagram ser ud lige i forhold til flowet i hvordan programmet fungerer, hænger sammen og det overordnet funktionalitet af hele systemet.

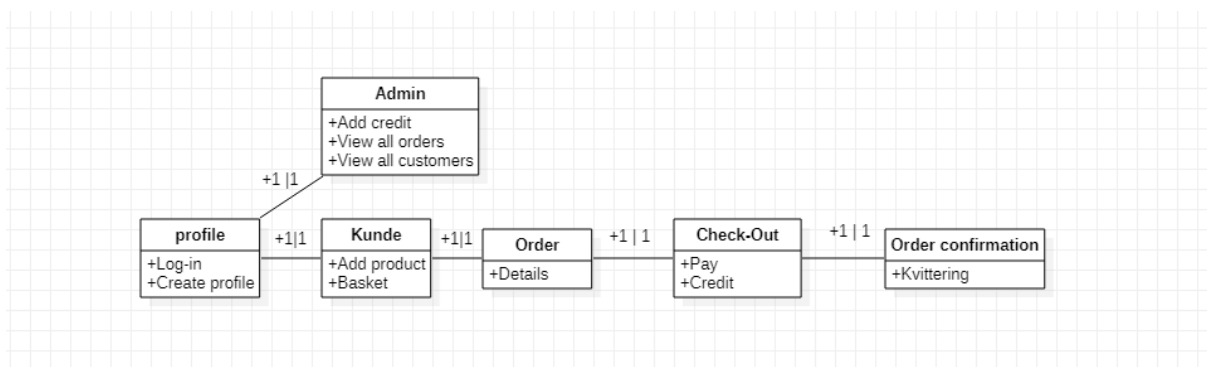
TO-BE:



Aktivitets diagrammet heroppe viser hvordan vores TO-BE diagram ser ud lige i forhold til hvordan vi vil ha vores program til at køre senere hen i den nærmeste fremtid.

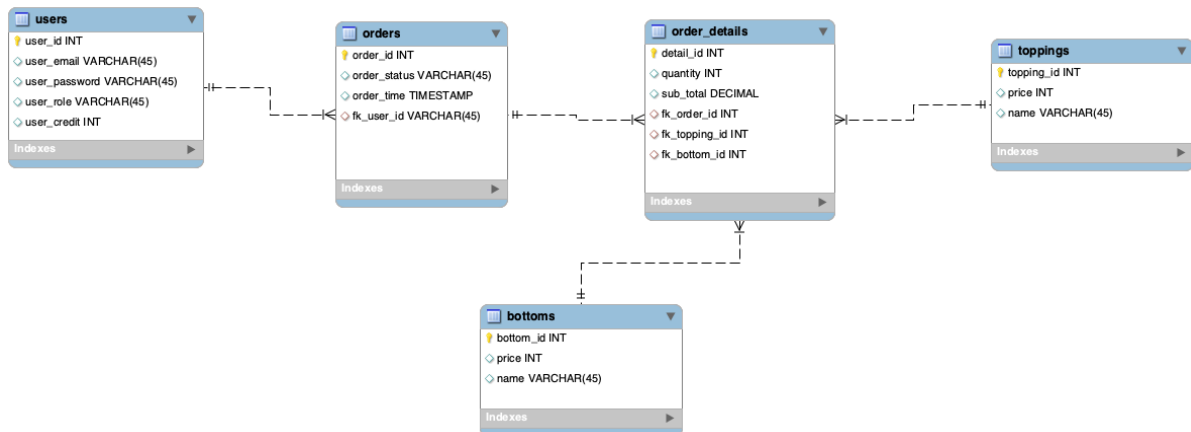
Domænenemodell og ER Diagram:

Domænenemodell:



Domæne modellen heroppe viser hvordan de relationer hænger sammen i vores system, og hvordan det fungerer i forhold til funktionaliteten af både hvordan kunderne bestilling af cupcakes i hele systemet.

ER Diagram:



- Hvis nogle af tabellerne ikke er på 3. normal form vil det være almindeligt at nævne det, og forklare hvorfor det er gjort (tidspres eller anden overvejelse).

Vores tabeller er ikke i 3. Normal form, fordi vi var ret presset på tid og havde overvejelser

- Hvis der anvendes 1-1 relationer kan man beskrive hvorfor man ikke blot har en tabel. Vi gøre ikke bruge af 1-1 relationer.

- Hvis nogle tabeller implementerer en mange-mange relation vil det være normalt at beskrive det.

Ja det er normalt til at beskrive hvis nogle tabeller implementerer mange til mange relationer, så det er nemmere for andre programmører til at forstå hvordan tabellerne hænger sammen.

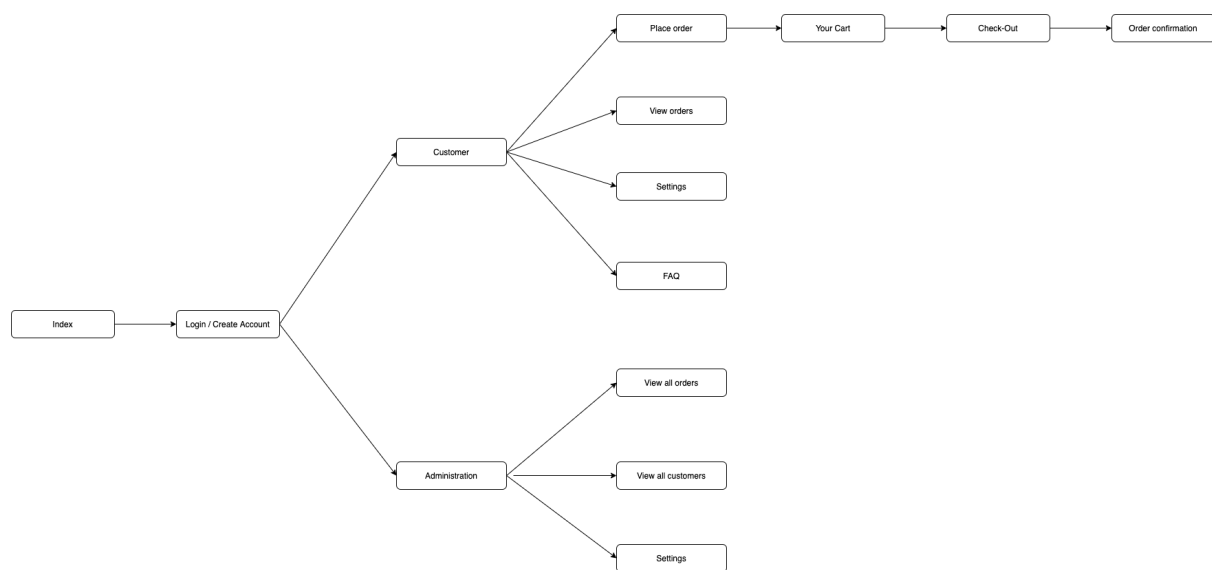
- Hvis der er flere veje at nå fra et sted til et andet, vil det nemt gøre det svært at holde databasen konsistent. Hvis I har gjort det alligevel så skal I forklare hvorfor.

Det er nemmere at være konsistent så man kan navngive sine kolonner og tabeller, så det giver mening, så du man kan holde styr på, hvad man laver.

- Hvis der er tabeller, hvor man benytter andet end et automatisk genereret ID som nøgle skal man forklare det.

Nej der er ikke nogle tabeller hvor vi benytter andet end automatisk genereret ID som nøgle, fordi meningen af ID'et er så den kan tælle op af sig selv.

Navigation Diagram:



Man kan kun bestille cupcakes hvis man har oprettet en profil som kunde, det samme gælder for hvis man vil se oversigt af alle sine order. Som admin kan man se oversigt af alle customers, samt den side hvor man kan se all order og indsætte credit inde i en kundes konto.

Særlige forhold:

- **Hvilke informationer gemmes i session**

Det der bliver gemt i session er indholdet af information fra useren.

- **Hvordan håndterer man exceptions. Det kommer vi tilbage til senere i semesteret.**

Med try catch

- **Hvordan man har valgt at lave validering af brugerinput.**

Man har valgt at lave validering af brugerinput ved at bruge "throw new UserException" som er brugt til at tilbagemeld et fejlbesked hvis man ikke har indtastede det rigtige information i input feltet.

- **Hvordan man har valgt at lave sikkerhed i forbindelse med login.**

Man har valgt at lave sikkerhed i forbindelse med login ved at bruge Command.java filen til

at bestemme hvem har adgang til hvilken side, kommer an på om hvis man er employee eller kunde.

- **Hvilke brugertyper, der er valgt i databasen, og hvordan de er brugt i jdbc**

De brugertyper vi har valgt i databasen er customer og employee, de er brugt i jdbc til at bestemme hvilke bruger har adgang til de forskellige sider.

Status på implementation:

- **Man har ikke nået at lave alle de jsp sider man har med i navigationsdiagrammet.**
Vi mangler alle jsp sider som tilhører redigering.
Vi mangler at lave funktionaliteten til create order.
- **Man har ikke nået at lave alle CRUD metoderne til alle tabellerne**
Nej vi har ikke nået at lave delete.
- **Man har ikke fået stylet sine sider**
Alle vores sider er stylet og ser alle sammen perfekt ud. (jon sagde det selv :)).
- **Man har fundet en fejl "i sidste øjeblik", men har ikke haft tid til at rette det. - F.eks. at man har brugt session forkert, sådan at man på en af siderne kan komme ind uden at være logget ind.**
Vi har stadig mangler. TBD.
- **tests der fejler på afleveringstidspunkt**
Vi mangler tests.

Process:

- **Hvad var jeres planer for teamets arbejdsform og projektforsløbet?**
Vi har i løbet af projektet siddet sammen på Discord og arbejdet sammen. Vi har til svære opgaver siddet allesammen og kigget på en af os streame en skærm løst opgaven i fælles gang. Når vi har haft mulighed for det, har vi fordelt opgaverne ud, mens vi stadig har siddet sammen på Discord, så vi hurtigt kunne hjælpe hinanden hvis vi sad fast.
- **Hvordan kom det til at forløbe i praksis?**
Det gik strålende og fremragende efter planen.
- **Hvad gik godt?**
Vores evne til at samarbejde og hjælpe hinanden når vi havde brug for det, samt vores dedikation at sidde sammen i lang tid hverdag og lave opgaven fællesskab. Brugen af github gik meget bedre i denne projekt sammenlignet med forrige projekter.
- **Hvad kunne have været bedre?**
Blive bedre til at kode når det drejer sig om Java og funktionaliteten i det.
- **Hvad har I lært af processen?**
Projektet har givet os god indblik i at hvad det kræver at oprette en Servlet og få en

nogenlunde hjemmeside oppe og køre. Det er blevet nemmere at bruge diagrammer og modeller til at få et overblik af projektet.

Vi er blevet bedre til databaser og at bruge MySQL samt implementere det i Java og projektet.

- **Hvad vil I evt. gøre anderledes næste gang?**

Ikke brug for lang på opgaven når det ikke er nødvendigt og bedre planlægning af tidsforbrug.

Kom hurtigere i gang med funktionalitet, så vi kan få en fungerende program.

Blive bedre til at kode!!!.