

# Temporalis rapport - Cupcakeshop



Deltagere: navn - cph email - github navn.

- |                           |  |               |
|---------------------------|--|---------------|
| • Magnus West Madsen      | - <a href="mailto:cph-mm454@cphbusiness.dk">cph-mm454@cphbusiness.dk</a> | - EmSSoH      |
| • Mads Heidemann Justesen | - <a href="mailto:cph-mj675@cphbusiness.dk">cph-mj675@cphbusiness.dk</a> | - MadsJuste   |
| • Kim Nowak               | - <a href="mailto:cph-kn191@cphbusiness.dk">cph-kn191@cphbusiness.dk</a> | - KimNowak47  |
| • Esben Fribo Gøtttsche   | - <a href="mailto:cph-eg42@cphbusiness.dk">cph-eg42@cphbusiness.dk</a>   | - EFGotttsche |

Klasse: B

# Indledning

Denne rapport er udarbejdet af 4 2. semester studerende på cphbusiness datamatikerstudie. Vi har fået udleveret opgaven på andet semester og alle krav til opgaven er blevet specificeret af lærerne.

Rapporten består af følgende:

Baggrund - kort om hvad opgaven går ud på.

Teknologi valg - hvilke programmer, versioner osv. der brugt i forbindelse med projektet

ER diagram

Navigationsdiagram

Sekvensdiagram

Status på implementation - en oversigt over implementationerne og kommentarer til tilstanden af dem.

# Baggrund

Projektet har i korte træk været at opsætte en webshop for en cupcake bager. Hjemmesiden skal primært bruges til at håndtere bestillinger for kunder.

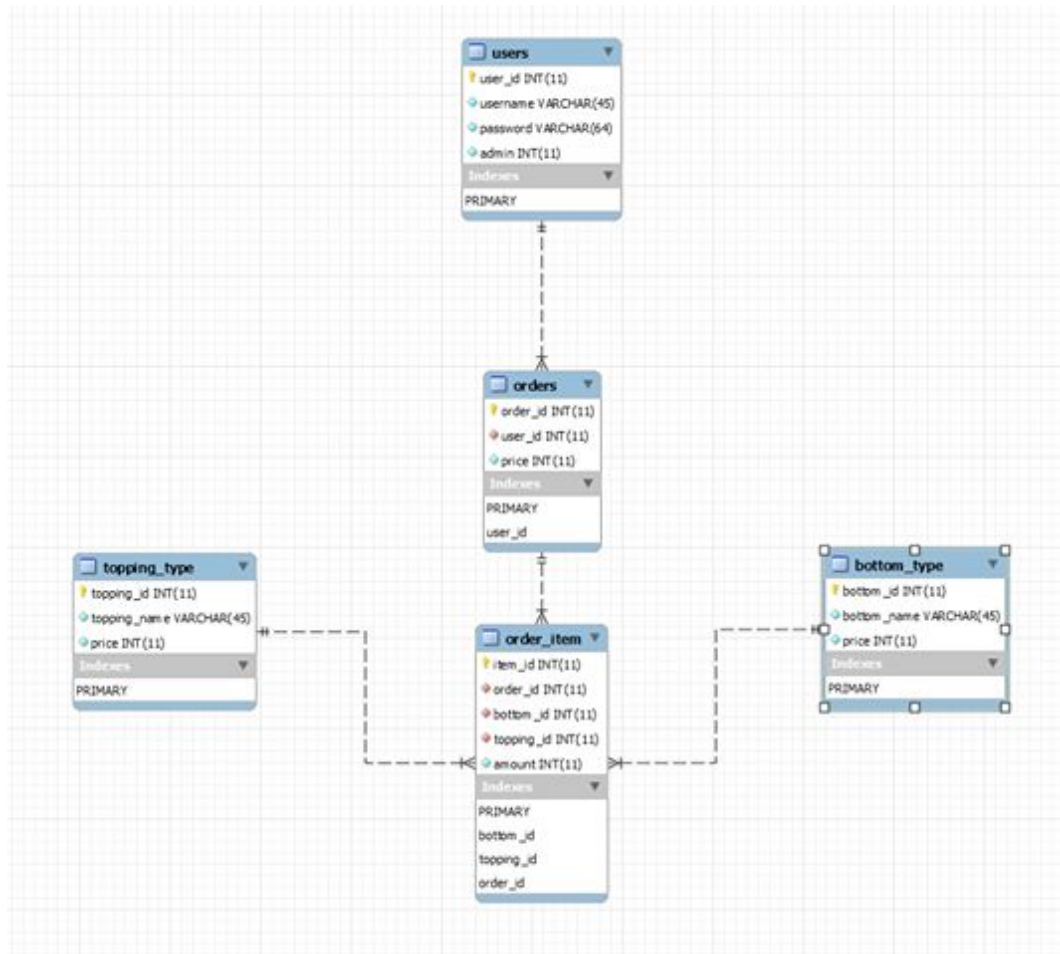
Man skal som kunde, kunne logge ind som en bruger. Derfra skal det være muligt at placere ordrer på cupcakes.

Vi fik angivet en liste af muligheder for henholdsvis bunden og toppen af af selve kagen kan laves af, og det skal så være muligt at sammensætte sin egen egen cupcake, samt vælge antallet man vil bestille.

# Teknologivalg

Netbeans 8.2	brugt til at skrive vores kode.
MySQL 6.3.9 CE	brugt til at lave vores database.
JDBC	Connector til databasen.
Git hub.	Versionsstyring.
Tomcat 8	Kontrol af servlets etc.
DigitalOcean	brugt til at bygge vores server
putty	SSH keys
Git Bash	push, pull fra online repository.

# ER Diagram

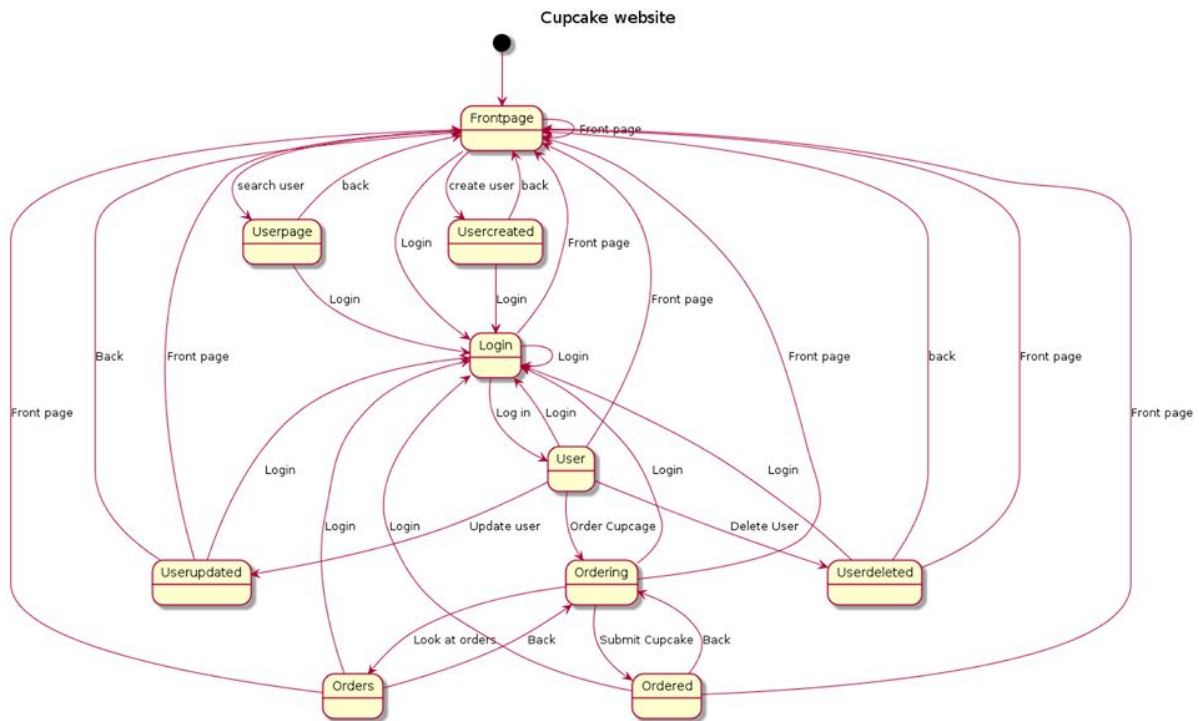


Dette er et diagram over vores database og den viser hvordan den er opsæt.

Vi følger den 3. normalform i hele vores database.

Alle tabeller bruger 1 til mange relation. Desuden har alle tabeller en primary key, orders og order\_item bruger begge fremmede nøgler. Orders bruger user\_id og order\_item bruger order\_id, bottom\_id og topping\_id

# Navigationsdiagram



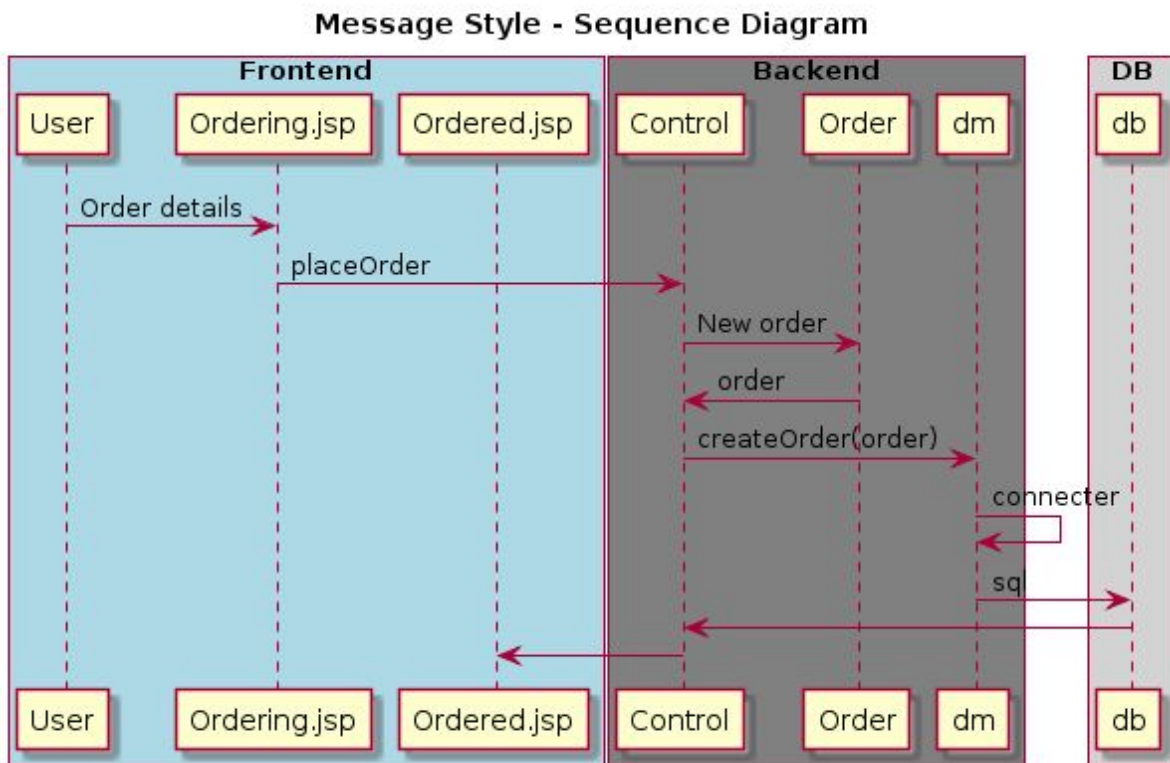
Teknisk set kan man komme til alle .jsp sidder på vores hjemmeside hvis man kender til navne på jsp filerne, man skal blot ændre url (ip/navn/jspnavn) og så sender den dig til siden.

Hvis man navigere hjemmesiden igennem selve siden kan man dog kun komme til user.jsp hvis man logger ind og fra user kan man komme til update/delete/order.

Vi bruger en fælles navigation bar på alle sider hvilket giver den rodet model da alle sider kan føre til Login og Frontpage.

Vi har kun en servlet (control) som håndterer alt transport mellem siderne.

# Sekvensdiagram



På dette diagram ser vi hvordan vores program reagerer på en bruger når de forsøger at oprette en ordre.

Først indtaster de order details, som bliver submitted på ordering.jsp og en request bliver behandlet i Control servleten. Denne servlet laver et order objekt, hvorefter den benytter createOrder(order) metoden i datamapper klassen, parameteren order er objektet den lige har oprettet. DataMapperen forsøger at åbne en connection til databasen hvorefter den sætter ordren ind i en tabel. Derefter bliver der sendt en boolean tilbage om det lykkedes eller ej, og brugeren bliver redirected til ordered.jsp, som er en side der viser at ordren er blevet placeret.

## Status på implementation

punkt	Status	information
Database	implementeret	Har information omkring User, orders, order_items, Topping og bottom
Login	implementeret	
User	implementeret	Kan create, update og delete users. deruden gå videre til bestilling
Bestil Cupcake	Delvist implementeret	Kan bestille, den sender dog ikke info til databasen
Admin	ikke implementeret	Der er Admin status men der mangler aktuelle funktion.
Indkøbskurv	Delvist implementeret.	Siden er lavet så den fungerer, men der er et problem med databasen
Login sikkerhed	ikke implementeret	Vi arbejde på hashing men har ikke ændre sikkerheds former
Jsp sider	implementeret	
CSS	implementeret	har 3 forskellige CSS filer. disse kunne dog nemt konverteres til en enkelt
Crud	Implementeret	Brugt alle
Orders	Delvist implementeret	Man kan ikke se hvilken ordre man har bestilt tidligere. Samme som kurven





