

Gruppe 8: Jonathan, Claus og Frederik

Cupcake



Github link: <https://github.com/Bringordie/CupcakeProject>

INDLEDNING

Formålet med opgaven er at programmere og dokumentere et onlinesystem, som kan håndtere ordre på cupcakes for et unavngivet bageri. Systemet skal effektivisere processen med at afgive ordrer, således at bageriets kunder ikke behøver at ringe eller møde fysisk op i butikken for at afgive bestilling. Hermed frigives også tid og energi for bageriets personale i forbindelse med ordremodtagelse. Bageriets kunder skal via loginoplysninger have adgang til en side, hvor de kan vælge hvilken kombination af topping og bottom, der skal udgøre den ønskede cupcake. Derudover skal de kunne angive antallet af den ønskede cupcake. For at kunne logge ind, skal kunden kunne registrere en bruger, som gemmes i en database. Når kunden er færdig med at afgive bestilling, skal vedkommende stilles videre til en indkøbskurv, der angiver den samlede bestilling samt den totale pris på bestillingen.

Ønsket er mao, at systemet skal kunne;

1. registrere bruger (gennem oplysninger om brugerens ønskede brugernavn, password, e-mail, adresse mv.).
2. brugerens loginoplysninger gemmes herefter i en database, så vedkommende har mulighed for at logge ind
3. Hvis brugerens indtastede brugernavn og password matcher en tilsvarende kombination i databasen, skal brugeren have adgang til en JSP-side, hvor det er muligt at bestille cupcakes (hvis ikke skal en fejlmeddelelse poppe op, og brugeren forbliver på loginsiden.
4. Hver cupcake består af en topping og en bottom, som brugeren selv skal vælge kombinationen på. Derudover skal brugeren også angive antallet af den bestemte cupcake, der ønskes. Bestillingen gemmes herefter i databasen.
5. Når brugeren er færdig med bestilling, skal det være muligt at gå videre til en JSP-side, der viser den totale bestilling samt den totale pris for bestilling.

Rent praktisk ønsker vi at lave en database, der indeholder data på de forskellige varianter, der er inden for hhv. topping og bottoms. Derudover skal databasen også gemme data på brugerens bestilling, samt indeholde data på brugerens oplysninger i forbindelse med login. Dertil skal vi i netbeans lave et datalag, der skaber forbindelse til vores database og henter den ønskede data.

Hertil skal vi have et logiklag, der står for de nødvendige udregninger samt et præsentationslag, der sørger for at kalde de bestemte JSP-sider. Det skal i den forbindelse siges, at gruppen har valgt at lave en servlet (frontcontroller) som håndterer en række JSP, herunder registrering, login, shop og shoppingcart.

I forhold til rapportens opbygning gør vi først rede for hvilke programmer og teknologier, vi har gjort brug af i forhold til at skabe systemet. Herefter følger en række diagrammer, der bl.a. har til opgave at danne et overblik over, hvordan vi har valgt at bygge systemet op. Til sidst vil vi redegøre for implementation.

Baggrund

Virksomheden som skal bruge systemet er en webshop der sælger cupcakes, dog kun til afhentning. Til hver cupcake hører en bund og en topping, der er adskillige valgmuligheder og de kan blandes vilkårligt, dog kan der kun være én topping og en bund i en cupcake.

Virksomheden har en meget hurtig bagemaskine og cupcakesene er derfor klar med det samme, efter de er bestilt og betalt. Betaling foregår via et optanknings system, hvor indbetaling skal ske manuelt i databasen til en bruger, som så bliver fratrasket, så snart der foretages køb på hjemmesiden. Dette betyder også, at virksomhedens kunder skal have mulighed for at oprette sig som bruger i systemet.

Teknologi valg

Vi bruger en Droplet fra <https://cloud.digitalocean.com/>

Netbeans version 8.2 (Java EE) <https://netbeans.org/downloads/8.2/>

Da vi bruger 8.2 Netbeans bruger vi den nyeste opdatering af Tomcat 8.5

<https://tomcat.apache.org/download-80.cgi>

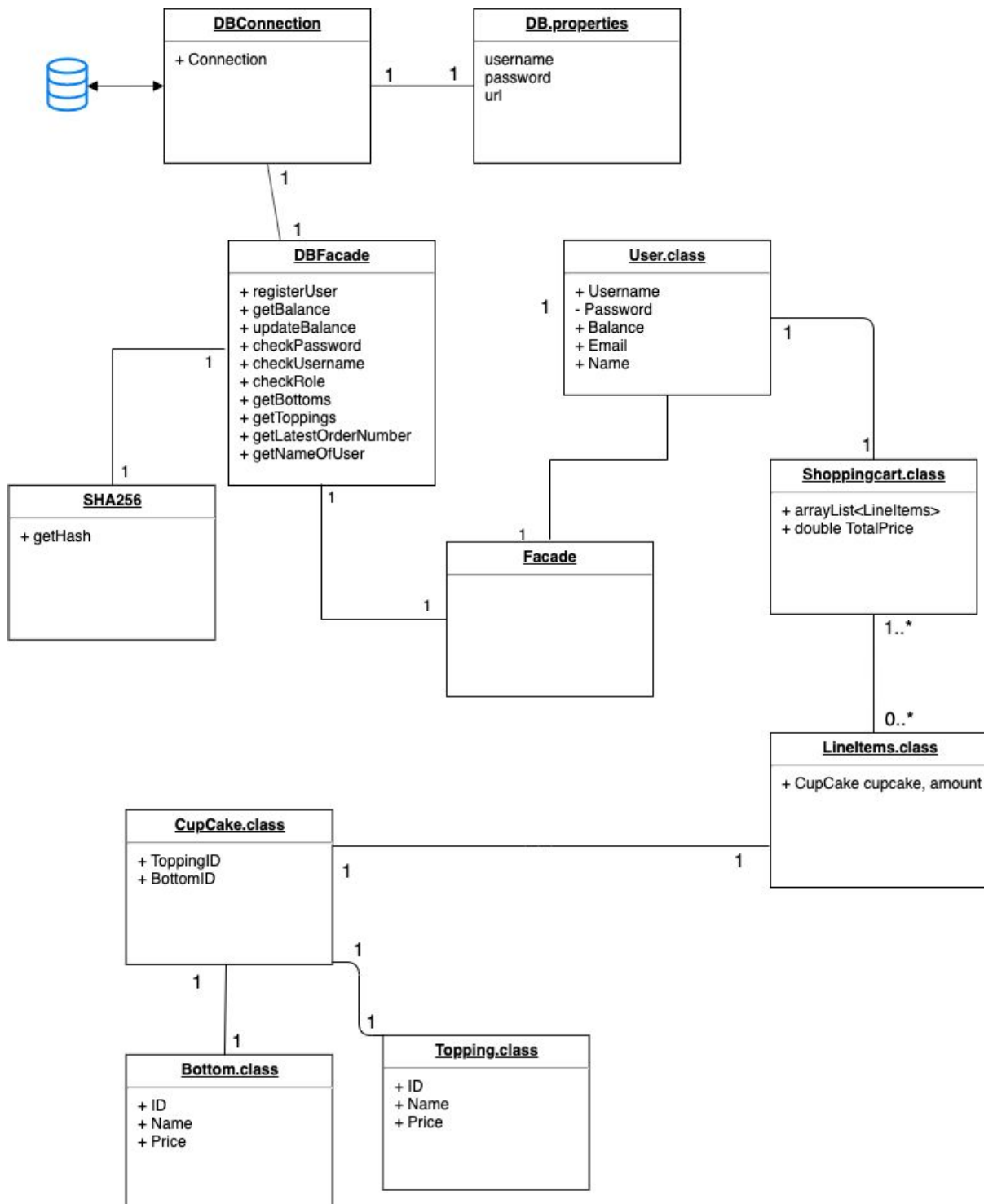
Github projektet ligger på <https://github.com/Bringordie/CupcakeProject.git>

Vi bruger JUnit version 1.4.1 i Netbeans.

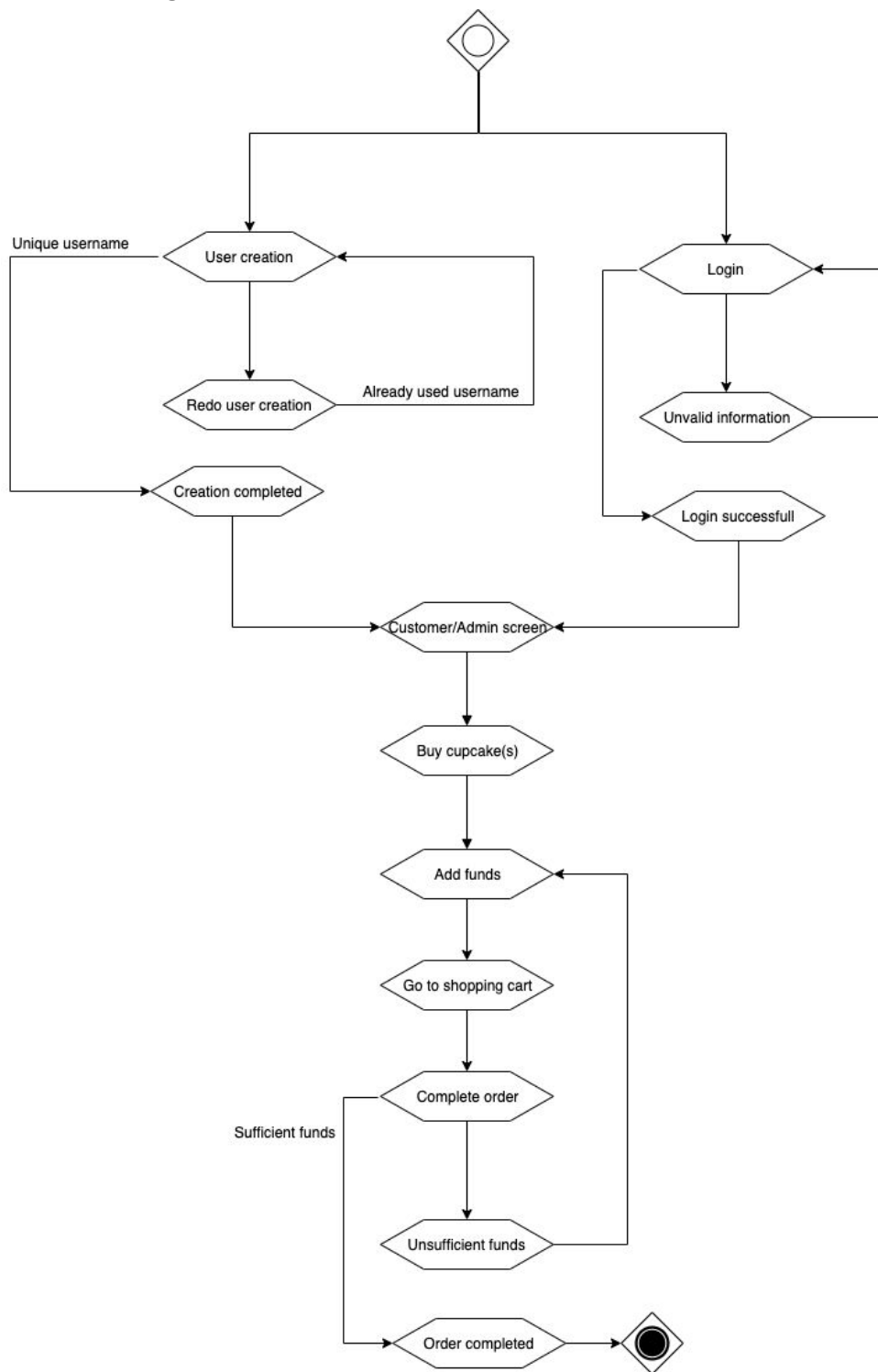
Til at se vores SQL data lokalt bruger vi den nyeste version af MySQL Workbench

<https://www.mysql.com/downloads/>

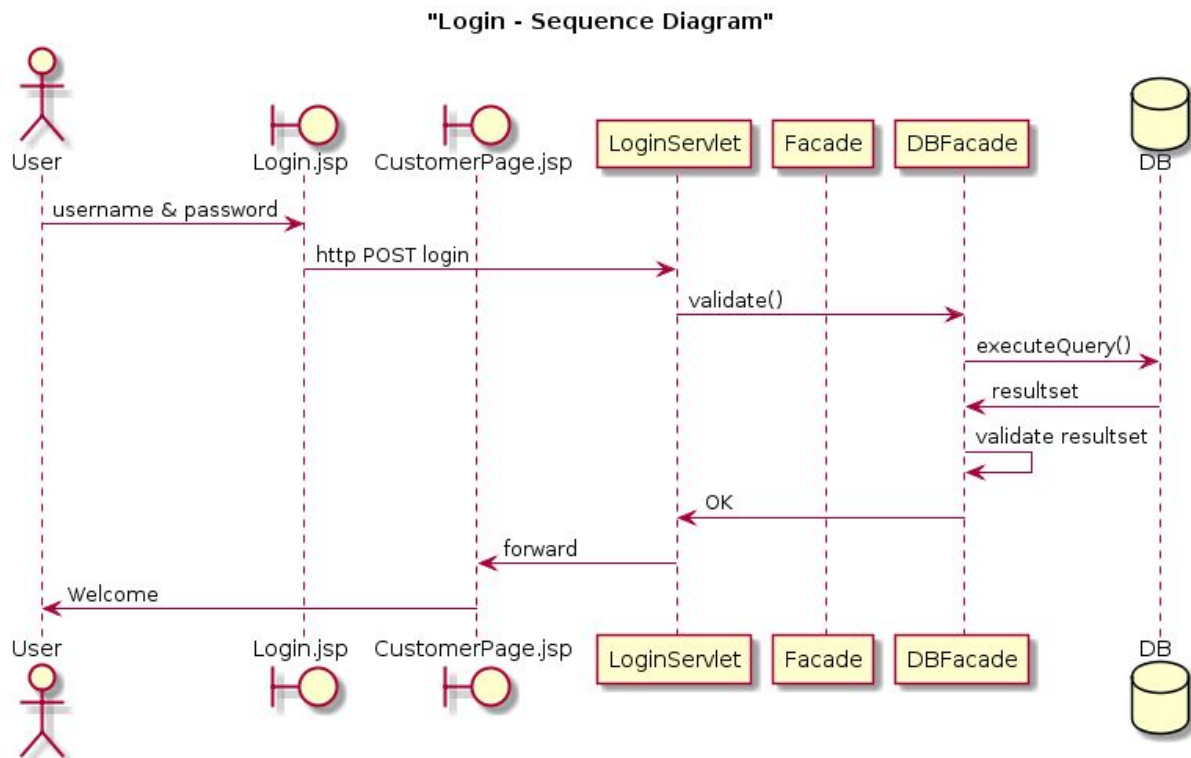
Domain model:



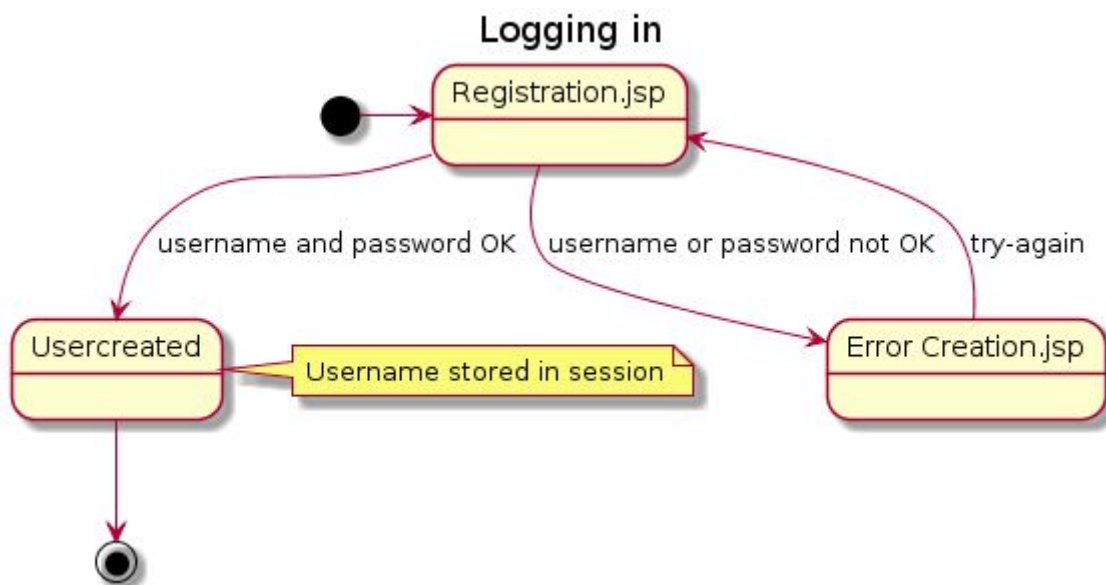
Aktivitets diagram:



Sekvens diagram:



Tilstands diagram:



ER diagram:

