

# CDIO-2

## Gruppe 11

10/4-2020

02324 - Videregående Programmering



Mikkel Danielsen  
s183913



Frederik Koefoed  
s195463



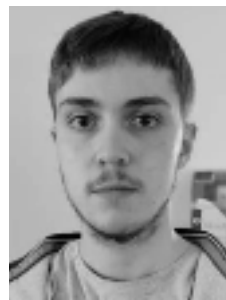
Muhammad Talha Butt  
s195475



Volkan Isik  
s180103



Lasse Strunge  
s195486



Mark Rune Mortensen  
s174881

# Indholdsfortegnelse

<b>Kravliste</b>	<b>2</b>
<b>Analyse</b>	<b>3</b>
UseCase Diagram	3
Use case beskrivelse	4
Fully dressed	4
Domæne model	7
Systemsekvensdiagram	8
<b>Design</b>	<b>9</b>
Design Class Diagram	9
<b>Dependency diagram over web</b>	<b>10</b>
<b>Implementering</b>	<b>11</b>
Overvejelser	11
3-lags modellen	11
Database server	11
<b>Demodata</b>	<b>12</b>
<b>Brugervejledning</b>	<b>13</b>
<b>Test</b>	<b>14</b>
<b>Bilag</b>	<b>15</b>

# Kravliste<sup>1</sup>

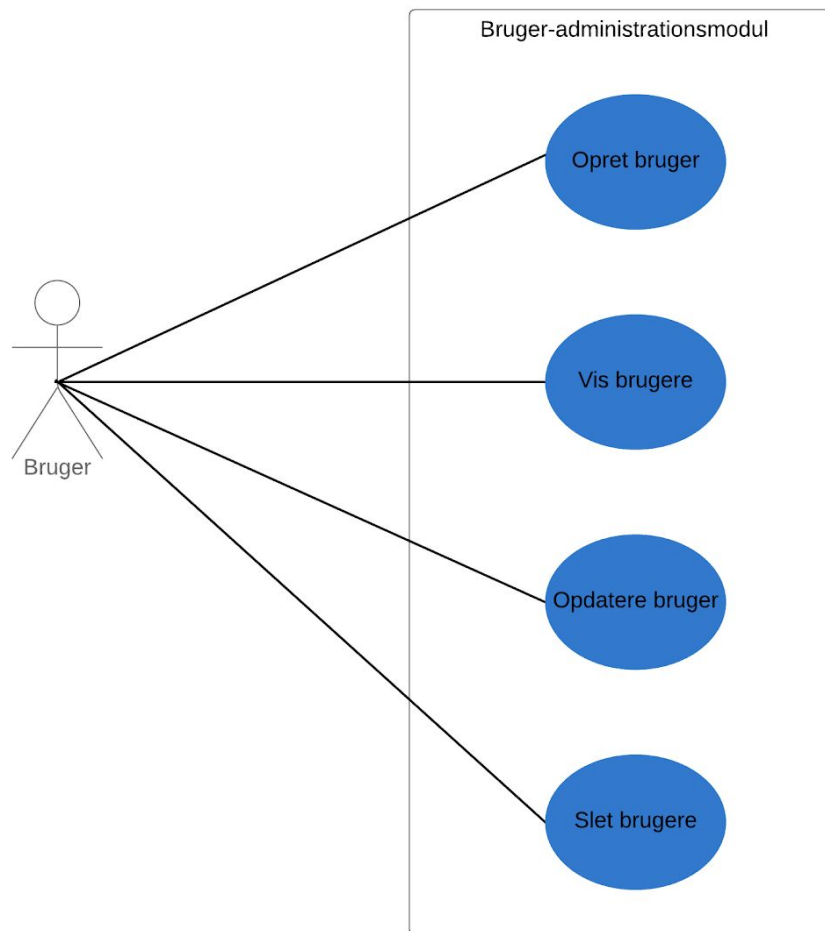
- K1 - Programmet skal styres af en single page application der tilgås gennem en webbrowser
- K2 - Single page applicationen skal implementeres med en REST-backend og en HTML/CSS/Javascript frontend.
- K3 - Det skal være muligt at oprette en bruger
- K4 - Når man opretter en bruger skal man kunne angive deres navn, initialer, cpr-nummer samt deres roller
- K5 - Man skal kunne se data for alle oprettede brugere på en samlet liste
- K6 - Brugerens ID skal være mellem 11 og 99
- K7 - Man skal kunne ændre i brugerens oplysninger ( Navn, initialer, cpr-nummer, password)
- K8 - Man skal kunne slette brugere
- K9 - Der er 4 roller (Admin, Pharmacist, Foreman, Operator)
- K10 - Brugernavnet skal være mellem 2 og 20 karakterer
- K11 - Initialerne skal være mellem 2 og 4 tegn
- K12 - Passwordet skal indeholde mellem 6 og 50 tegn
- K13 - 3 af 4 følgende kategorier skal opfyldes for passwordet: der skal være 1 småt bogstav, der skal være 1 stort password, der skal være 1 cifer fra 0-9, der skal være et specialtegn ({',', '-', '\_', '+', '!', '?', '='})
- K14 - Alle metoderne i interfacet IUserDAO skal implementeres
- K15 - Når en ny bruger oprettes skal der genereres et password til dem
- K16 - Når en ny bruger oprettes skal de have det laveste ledige ID
- K17 - Programmet skal kunne afsluttes fra menuen

---

<sup>1</sup> Bilag 1

# Analyse

## UseCase Diagram



## Use case beskrivelse

### Fully dressed

Use Case Section	Comment
Navn	Opret bruger
Scope	CDIO 2
Level	User-goal
Primær aktør	Bruger
Preconditions:	Hjemmesiden er startet og brugeren er logget ind.
Postconditions:	En ny bruger er oprettet
Hoved succes scenarie:	<ol style="list-style-type: none"><li>1) Brugeren trykker på knappen "Opret Ny Bruger"</li><li>2) Navn bliver indtastet.</li><li>3) Initial bliver indtastet.</li><li>4) CPR-nummer bliver indtastet.</li><li>5) Password bliver indtastet.</li><li>6) En rolle eller flere roller vælges fra menuen.</li><li>7) Brugeren trykker på knappen "Opret".</li><li>8) Brugeren bekræfter oprettelse ved at trykke på "ok".</li><li>9) Brugeren bliver oprettet.</li></ol>
Alternative scenarier:	<p>2-6: Brugeren ikke indtaster på nogle af de felter, 7: Brugeren trykker på knappen "Opret". 8: Brugeren får fejlmeddelelse at den skal udfylde de tomme felter og bruger trykker på "ok", og udfylder de tomme felter. 7: Brugeren trykker på knappen "Opret". 8a: Brugeren bekræfter oprettelse ved at trykke på "ok". 8b: Brugeren trykker på "cancel", der sker ikke noget man bliver på samme side.</p> <p>2-6: Brugeren indtaster alle oplysninger 7: Brugeren trykker på knappen "Opret". 8a: Brugeren trykker på "cancel" og der sker ikke noget man bliver på samme side. 8b: Brugeren trykker på "ok" og brugeren bliver oprettet.</p>

Use Case Section	Comment
Navn	Vis brugere
Scope	CDIO 2
Level	User-goal
Primær aktør	Bruger
Preconditions:	Hjemmesiden er startet
Postconditions:	Liste af brugere bliver vist
Hoved succes scenarie:	<ol style="list-style-type: none"> <li>1) Brugeren indtaster brugernavn og password.</li> <li>2) Brugeren trykker på "login"</li> <li>3) Brugeren er logget ind.</li> <li>4) Brugeren kan nu se alle de oprettede brugere i systemet.</li> </ol>

#### Fully dressed

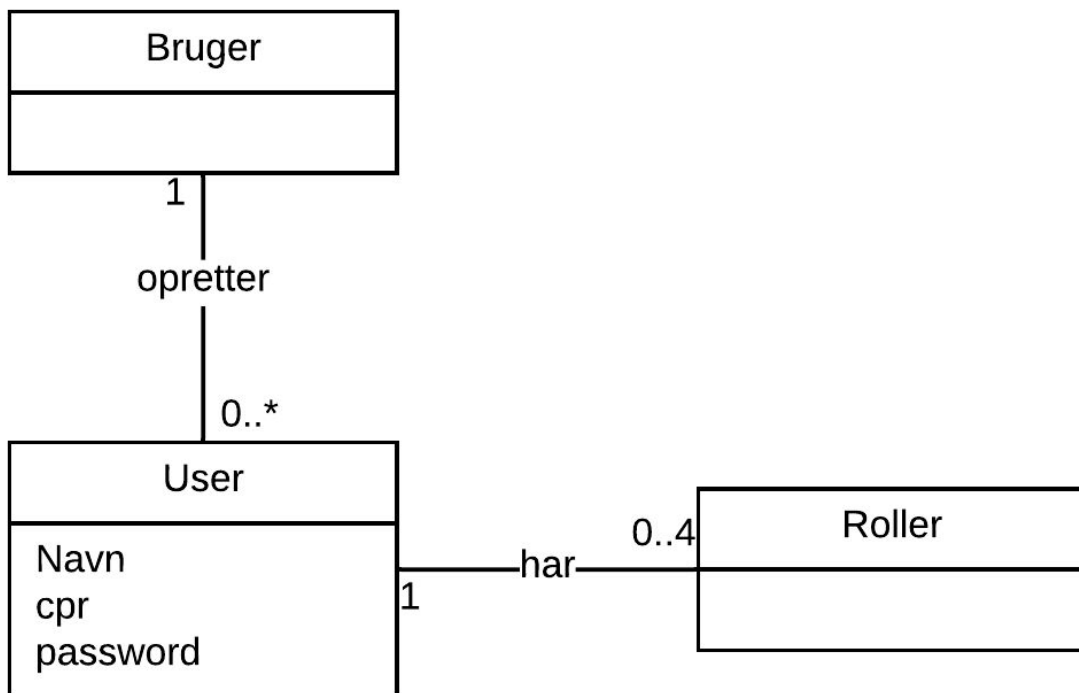
Use Case Section	Comment
Navn	Opdater bruger
Scope	CDIO 2
Level	User-goal
Primær aktør	Bruger
Preconditions:	Programmet startet og en database valgt
Postconditions:	En bruger bliver ændret
Hoved succes scenarie:	<ol style="list-style-type: none"> <li>1) Brugeren trykker på knappen "update" bruger.</li> <li>2) Brugeren indtaster ændringerne som skal ændres (Brugernavn, initialer, cpr, password, role).</li> <li>3) Brugeren trykker på knappen "update".</li> <li>4) Brugeren bekræfter sine ændringer og trykker "ok".</li> </ol>
Alternative scenarier:	<p>4.a: Brugeren trykker på "OK" og nu er brugeren har ændret sine oplysninger.</p> <p>4.b: Brugeren trykker på "CANCEL" og opdater siden bliver gentaget igen.</p>

--	--

### Fully dressed

Use Case Section	Comment
Navn	Slet bruger
Scope	CDIO 2
Level	User-goal
Primær aktør	Bruger
Preconditions:	Hjemmesiden startet og en brugeren er logget ind
Postconditions:	En brugere er blevet slettet
Hoved succes scenarie:	<ol style="list-style-type: none"> <li>1. Brugeren trykker på "slet" knappen.</li> <li>2. Brugeren får en besked på toppen af skærm, om de sikker på at slette den bruger.</li> <li>3. Brugeren trykker på "ok"</li> <li>4. Den valgte bruger bliver slettet fra systemet</li> </ol>
Alternative scenarier:	<ol style="list-style-type: none"> <li>1. Brugeren følger trin 1-2 fra hoved succes scenarie.</li> <li>3.a: Brugeren trykker på "cancel" og der sker ikke noget</li> <li>3.b: Brugeren trykker på "ok", og brugeren er slettet fra system.</li> </ol>

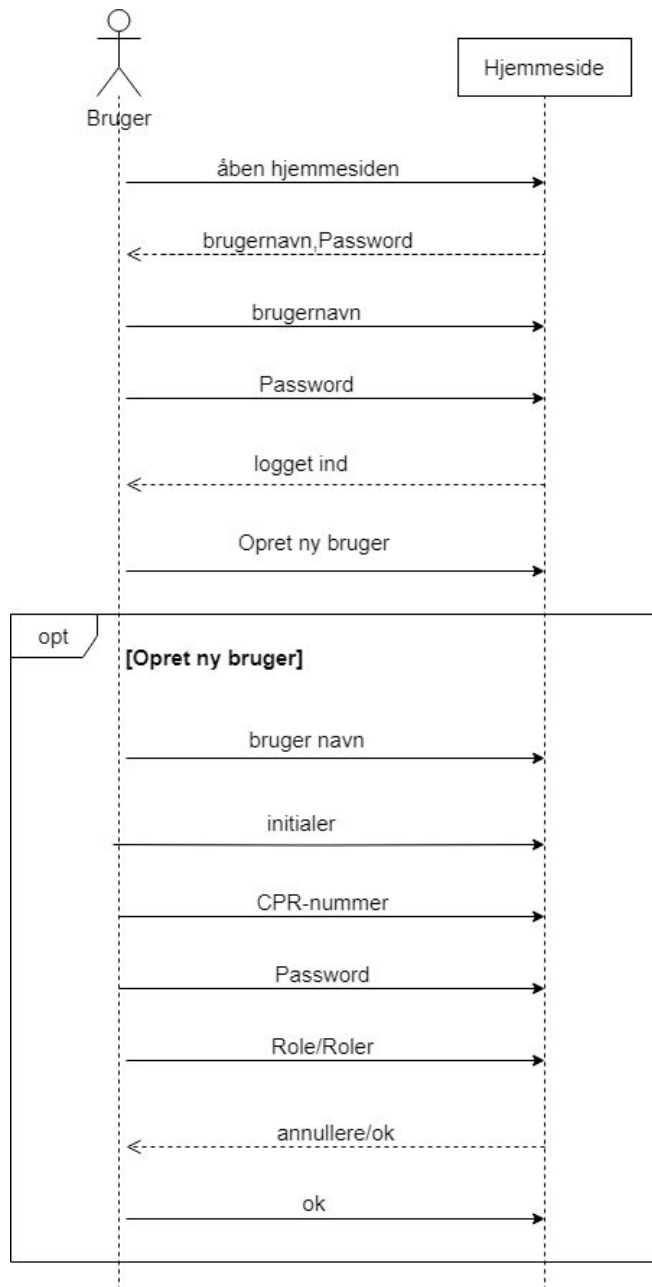
## Domæne model



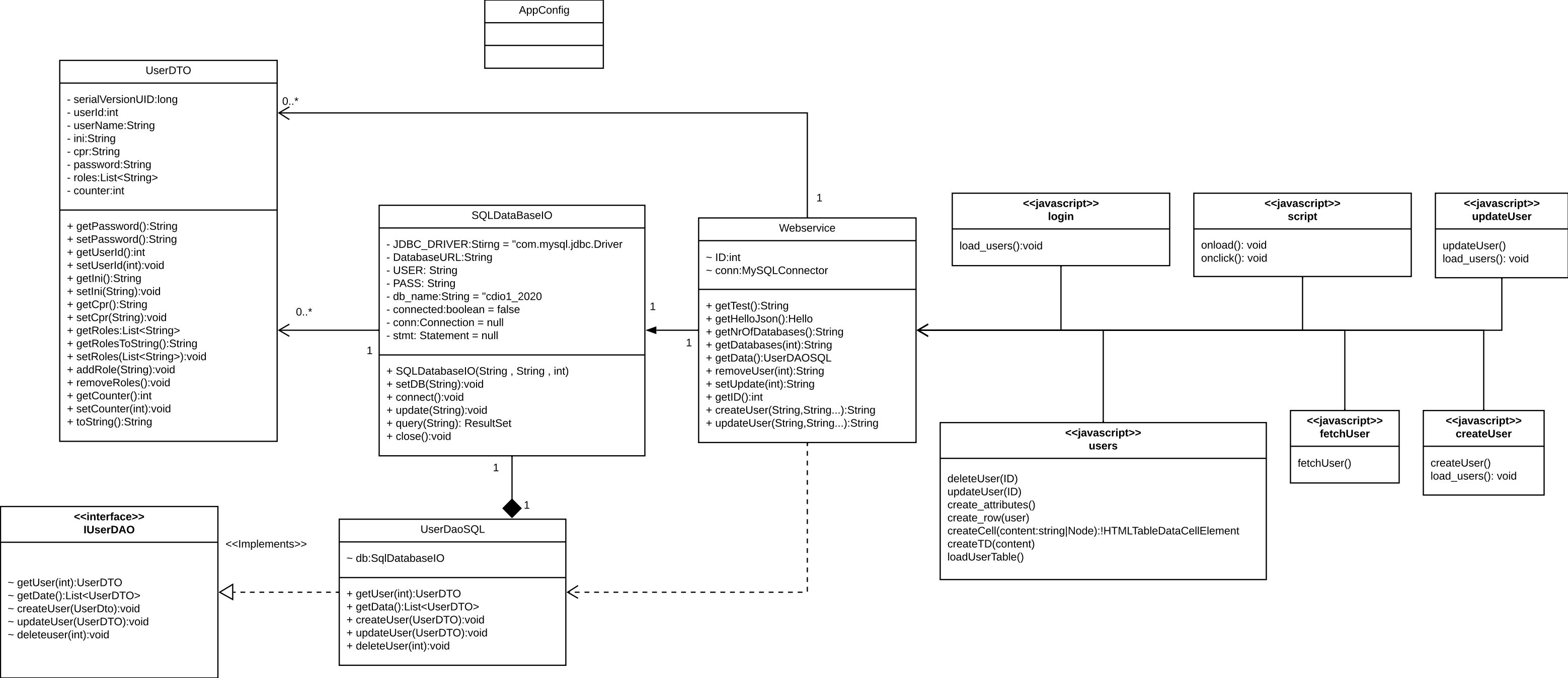


# Systemsekvensdiagram

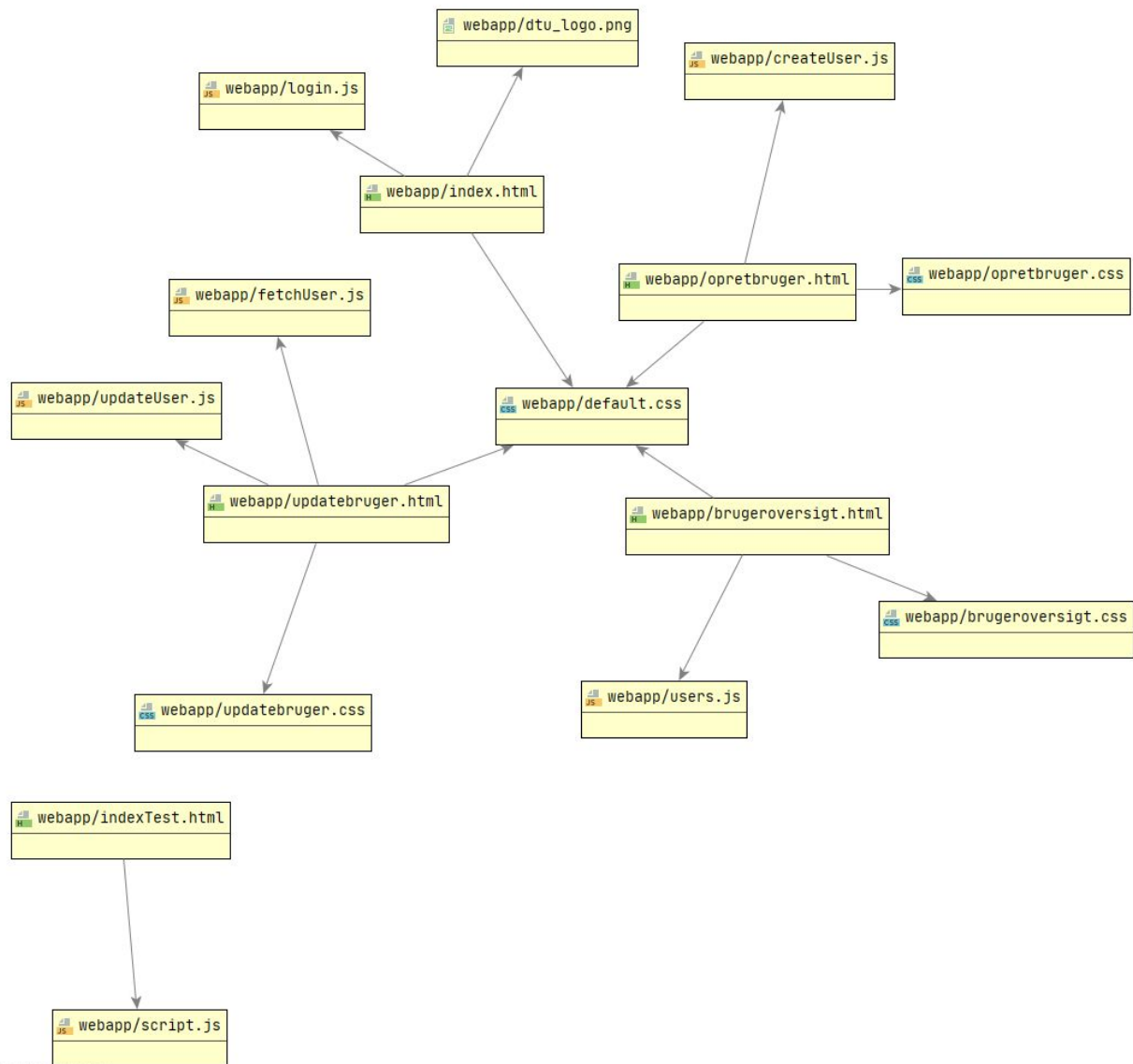
Nedenfor ses et systemsekvensdiagram over use casen "opret bruger":



# Design klassediagram



# Dependency diagram over web



Powered by yFiles

# Implementering

## Overvejelser

### 3-lags modellen

Vi har i vores projekt valgt at benytte os af trelagsmodellen, for at gøre systemets struktur mere overskuelig og de enkelte moduler nemmere at forstå. Samtidig bidrager modulariseringen til et system der er mere genanvendeligt, nemmere at vedligeholde samt lettere at udvide i fremtiden, da de enkelte delelementer kan betragtes isoleret.<sup>2</sup>

Vores grænseflade findes i webapp mappen og består af en række .html, .css og .js filer. HTML og CSS styrer strukturen og layoutet på hjemmesiden og JavaScript står for kommunikation med de forskellige webservices. I en fremtidig implementering, kunne dette med fordel bliver gjort vha. AJAX og JSON.

Vores controllerlag indeholder de forskellige webservices, der er dekorerede med vha. Jersey til at definere forskellige resources og actions. Vha. Jackson kan vi nemt oversat JSON til klassiske Java-objekter og omvendt.

Til sidst har vi datalaget, som controllerlaget arbejder tæt sammen med. Her finder vi de nødvendige klasser for at oprette en forbindelse til og kommunikere med vores database. Bemærk hvordan vi til at overføre information mellem de forskellige lag benytter en såkaldt DTO (data transfer objekt), til at overskuelige processen.

Som det kan ses, indeholder datalaget et interface (IUserDAO), hvilket vi med fordel også kunne have anvendt i de andre lag. Dette ville have forstærket effekt af modulariseringen yderligere og gjort vores software mere fleksibel.

Da der i vores løsning kun forekommer rene datamanipulationsopgaver, var der ikke har behov for et funktionalitetslaget. Skulle vi også have implementeret f.eks. kodeordsgodkendelse, havde det været en fordel med et funktionalitetslag. Bemærk også, hvordan vi benytter en såkaldt DTO (data transfer objekt) til at overføre information mellem de forskellige lag, for at overskueliggøre processen.

### Database server

Vi har i vores løsning valgt at benytte en online MYSQL server som database, for at undgå at brugeren skal have en lokal database server liggende. Databasen er opsat med tre tabeller: userdto (brugeren), roles (roller) og userdto\_roles (Hvilke userdto brugere har hvilke roller).

---

<sup>2</sup> "3 lags modellen" af Mads Nyborg og Uffe Kofoed, 2013, s. 1-2.

**userto**

userID	userName	ini	cpr	password
--------	----------	-----	-----	----------

**roles**

role_name
-----------

**userto\_roles**

userID	role_name
--------	-----------

## Demodata

Nedenfor ses den demodata som alle systemets databaser er blevet initialiseret med

ID	Navn	Ini.	CPR	Kodeord	Roller
11	Volkan	VOL	0912941235	SkiftMig!	Admin
12	Mikkel	MIK	3007902345	SkiftMig!	Pharmacist
13	Mark	MAR	1603783455	SkiftMig!	Foreman
14	Talha	TAL	2301974565	SkiftMig!	Operator
15	Frederik	FRE	0501025675	SkiftMig!	Pharmacist, Operator
16	Lasse	LAS	1104966785	SkiftMig!	Admin, Pharmacist, Foreman, Operator

# Brugervejledning

## Installation og kørsel af CDIO1.jar

- Programmet kræver Java 8.
- Programmet er bygget til at køre på en Windows 10 maskine.
- Apache Tomcat 8.5.54 Core downloades [her](https://tomcat.apache.org/download-80.cgi)<sup>3</sup> og pakkes ud.
- Udpak 11\_del2.zip.
- Mappen udpakket åbnes som projekt i IntelliJ Ultimate edition 2020.1.1+
- Mavenfilen pom.xml genimporteres
- Øverst til højre vælges "Edit Configuration"
- Tryk på plus tegnet øverst til venstre i vinduet og Og tomcat > local vælges.
- Application server Configureres ved at vælge mappen apache-tomcat-8.5.54
- Hold alt andet standard of tryk på "fix" nederst til højre.
- Her vælges artifaktet "BoilerPlate:war exploded" og der trykkes "Apply" og "OK"
- Programmet køres ved tryk på den grønne pil øverst til højre.

---

<sup>3</sup> <https://tomcat.apache.org/download-80.cgi>

# Test

<b>Test case ID</b>	TC01
<b>Summary</b>	Tester om data på websiden bliver oprettet korrekt
<b>Procedure</b>	Klik opret ny bruger Skriv brugernavn: Bo Peter Skriv initialer BP Skriv CPR nummer 0102030405 Skriv password:test123 Vælg rolle Admin Tryk opret Kontrollér at det rigtige data står i de rigtige felter
<b>Data</b>	Bo Peter BP 0102030405 Test123 Admin
<b>Status</b>	Passed
<b>Created by</b>	Lasse Strunge
<b>Creation date</b>	10/05/2020
<b>Last tested</b>	10/05/2020
<b>Environment</b>	IntelliJ IDEA 2019.2.3 (Ultimate Edition) Build #IU-192.6817.14, built on September 24, 2019 on Windows 10 Enterprise Version 1903 OS build 18362.418 Google Chrome

<b>Test case ID</b>	TC02
<b>Summary</b>	Tester om slettefunktionen virker
<b>Procedure</b>	Klik slet bruger Tryk OK Kontrollér at brugeren ikke er der længere
<b>Data</b>	

<b>Status</b>	Passed
<b>Created by</b>	Lasse Strunge
<b>Creation date</b>	10/05/2020
<b>Last tested</b>	10/05/2020
<b>Environment</b>	IntelliJ IDEA 2019.2.3 (Ultimate Edition) Build #IU-192.6817.14, built on September 24, 2019 on Windows 10 Enterprise Version 1903 OS build 18362.418 Google Chrome

## Bilag

Bilag 1: CDIO-D1 Rapport:

[https://docs.google.com/document/d/1a9eWnNCtYzhWlhl8TpB49kwtZdvvW\\_2mMQBvW\\_alCTg/edit?usp=sharing](https://docs.google.com/document/d/1a9eWnNCtYzhWlhl8TpB49kwtZdvvW_2mMQBvW_alCTg/edit?usp=sharing)