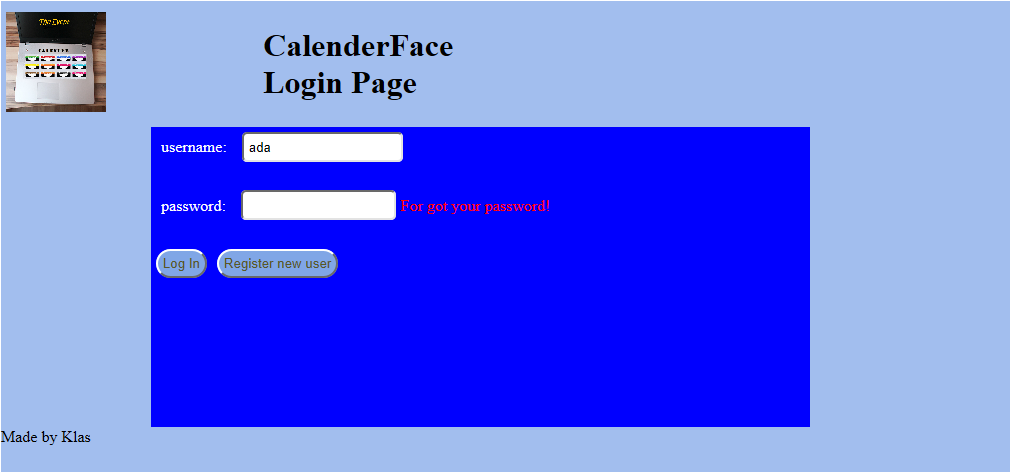
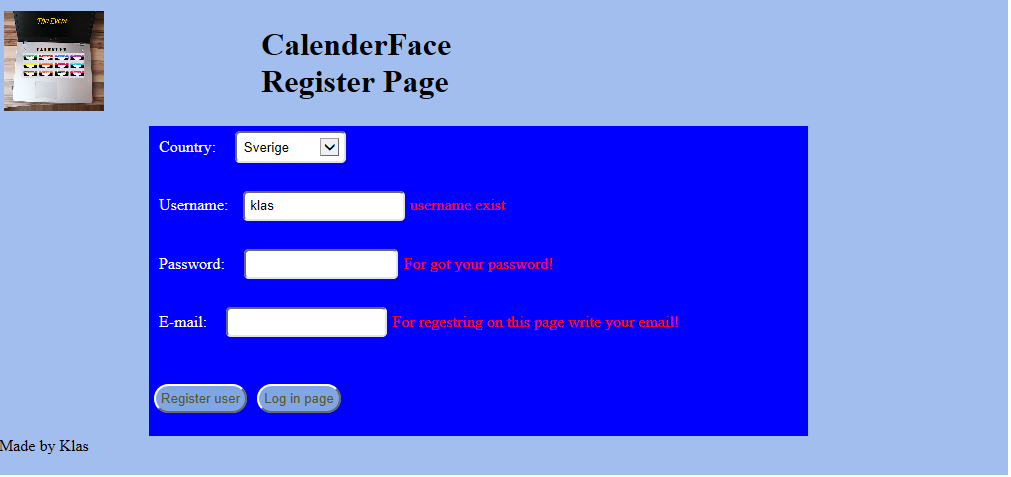
****Project Face Calender

I projektet har jag använt:

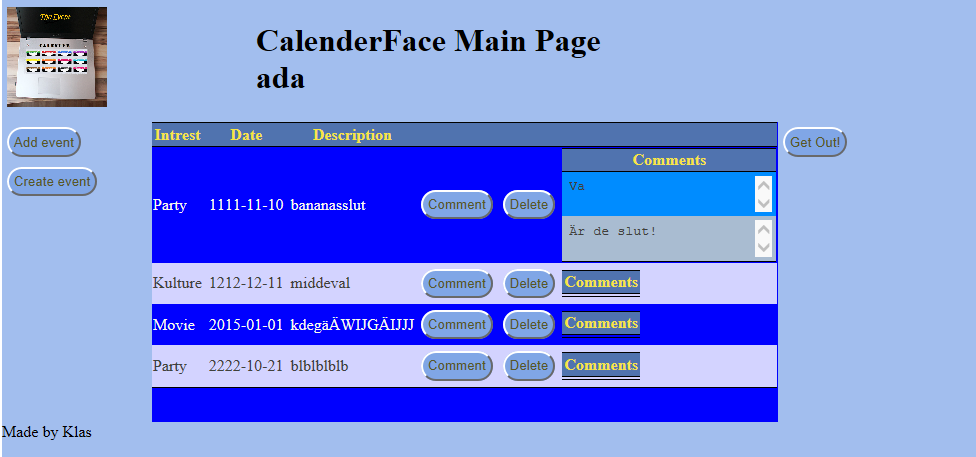
HTML5  
CSS3  
Ajax  
JSF2.2  
Wildfly 9.01  
Eclipse



Inloggningen sker på Index.xhtml.  
Lösenordsfältet får ej vara tomt, detta löser jag med hjälp av validation. Vid fel inloggningsuppgifter dirigeras man om till error.xml med hjälp av faces-config.xhtml och om rätt till main.xhtml.



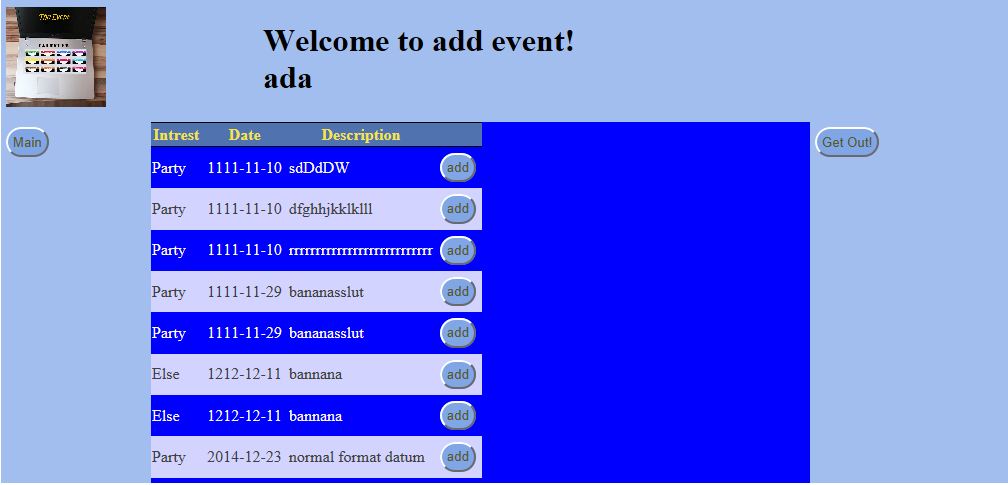
Registrering sker på register.xhtml. Fälten är country, username, password och e-mail. Country väljer man med en dropdown list. Det finns fyra olika alternativ men det går bara att välja en. Username finns det två kontroller på dels längden och om användarnamnet är upptaget. Det sker on blur med hjälp av ajxx. Jag har sett att de behövs trim innan man kontrollerar användarnamnet för att få namnet unikt. Passwordfältet får inte vara tomt någon annan kontroll har jag inte gjort på detta fält. E-mail kontrolleras mot ett mönster. Detta sparas i databasen user när man klickar på knappen registrera. Nästa steg är att få userid och beroende av utfall, dirigeras man om till main.xhtml eller errorxhtml med hjälp av faces-config.xhtml.



Main.xhtml är huvudsidan där man visar valt flöde. Varje rad består av intresse, datum, beskrivning samt två knappar comment och delete. Comment dirigeras vidare till commernt.xhtml och själva kommentarerna blev en textbox. Delete är inte implementerad på grund av tidsbrist.



Createevent.xhtml skapar eventen och den enda kontrollen är datumformatet. När eventet är skapat lagras det i databasen event.

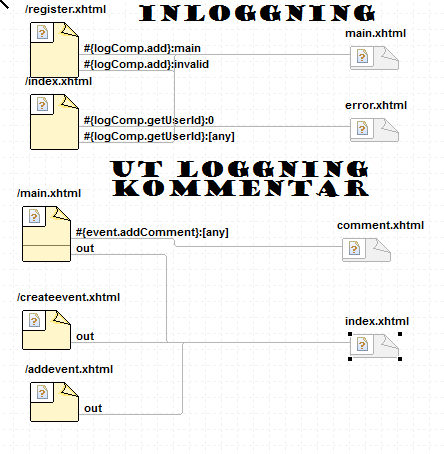


Addevent.xhtml visar event som användaren inte valt, sorterat efter datum. Väljer användaren att klicka på add kommer detta event att sparar i userevent databasen



När man valt att kommentera event kommer man till Comment.xhtml och man kan helt enkelt kommentera eventet, detta sparas i comment databasen.

1faces-config.xhtml faces-config.xhtml

**Databas**

Det blev fyra tabeller:

**User** för att lagra användaruppgifter, på sätt och vis blir denna inblandad i de mesta i och med att det är id:t som används när man skall bestämma vad som skall visas.

**Event** här lagras själva eventet med datum och beskrivning samt id och skapare.

**Userevents** uppgift är att hålla koll på vad användaren har valt att följa det görs med hjälp av id från user, event samt ett unikt id för posten.

**Comment** har koll på kommentarerna. Här finns komment id, user id, event id samt själva kommentaren.

-- MySQL Workbench Forward Engineering

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `facecalender` DEFAULT CHARACTER SET utf8 ;

USE `facecalender` ;

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `facecalender`.`event` (

`eventid` INT(11) NOT NULL AUTO\_INCREMENT,

`date` VARCHAR(45) NULL,

`intrest` VARCHAR(20) NULL DEFAULT NULL,

`description` VARCHAR(200) NULL DEFAULT NULL,

`userid` INT(11) NULL DEFAULT NULL,

PRIMARY KEY (`idevent`))

CREATE TABLE IF NOT EXISTS `facecalender`.`user` (

`id` INT(11) NOT NULL AUTO\_INCREMENT,

`password` VARCHAR(20) NULL DEFAULT NULL,

`userName` VARCHAR(20) NULL DEFAULT NULL,

`country` VARCHAR(20) NULL DEFAULT NULL,

`email` VARCHAR(25) NULL DEFAULT NULL,

PRIMARY KEY (`id`))

CREATE TABLE IF NOT EXISTS `facecalender`.`comment` (

`commentId` INT NOT NULL AUTO\_INCREMENT,

`eventId` INT NULL,

`comment` VARCHAR(200) NULL,

`userId` INT NULL,

PRIMARY KEY (`commentId`))

CREATE TABLE IF NOT EXISTS `facecalender`.`userEvent` (

`iduserEvent` INT NOT NULL AUTO\_INCREMENT,

`userID` INT NULL,

`eventId` INT NULL,

PRIMARY KEY (`iduserEvent`))

**Java**

Det har blivit fyra olika javabeans men tanken var egentligen fem, men av någon anledning lyckades jag inte att få det att fungera med fler än fyra bönor. De kraschade totalt och först fattade jag ingenting men med lite metodiskt sökande kom jag fram till att de kraschade när jag introducerade den femte bönan där av kom kommentarer att hamna under event.

**LogCom.java** hanterar användarna och kommunikation mot user tabellen, så de är främst index och register som hanteras men även förfrågningar om id.

**DataAdd.java** hanterar det som skall visas i främst main och addevent. Här skapas listor för visning olika events och kommentarer.

**Event.java** hanterar eventen som skall lagras i tabellen event och comment.

**UserEvent.java** hanterar event som är valda av användaren och sparas tabellen userevent.

**Design**

Designen blev ungefär som jag tänkt. Jag hade dock velat ha en mall att bygga från men fick inte det att fungera, som jag ville, i början. Jag har inte lagt jätte mycket energi på css utan bara sett till att css uppfyller uppgiften. Detta då jag finner det svårt att veta om uppdateringarna slagit igenom.

**Utvecklingsmöjligheter**

Jag vill visa hjälpmeddelande när man hovrar över textfälten. En sida där man kan hantera de event som man skapat. Även att användaren skall kunna ta bort event från sin lista. Jag har börjat med delete-funktionen men inte fullföljt denna. Kanske skulle man meddela att saker har lyckats. Det finns också delarna som jag inte har hunnit programmera.

**Sammanfattning**

De känns som det blev ganska komplext genom att man hela tiden måste hålla koll på användarens id så jag har i princip använt mig av tre olika metoder att få tillgång till id:t i bönan. Alla html sidor har id:t men att sedan få detta till bönan har varit en utmaning. I ett fall gjorde jag ett dolt fält som jag kunde läsa i bönan. När jag hade en commandButton kunde jag skicka med id:t. Svårast var i main när listan på event skulle skapas. Det finns ingen knapp men tillslut hittade jag ett sätt att finna username och den vägen få tillgång till id:t. Det är från loggin och register som problemet skapas eftersom det inte finns något id:t där utan skapas precis innan man dirigeras om.

Jag tycker att jag hållit mig till det jag tänkte i uppgift 2, lite nyansskillnader om vad jag valt att visa som till exempel visas aldrig id:t för användaren. Vad som inte blev implementerat är möjligheten att selektera i olika tidsintervaller, möjligheten att ladda upp bilder finns inte och felmeddelande på olika språk. Orsaken till att de inte blev implementerat är tidsbrist. Trots att allt inte kommit på plats har jag lärt mig mycket och uppgiften har jag funnit trevlig.