

Booking system

Obligatorisk opgave C#



14. april 2021

Mathias Foldager Andersen

EAAA 19U DATAMATIKER

# WPF:

I WPF delen har jeg kørt det måske lidt anderledes end mange andre. Jeg har valgt at kode rigtig meget af systemet i C# kode fremfor at begå mig i XAML kode. Dette har jeg valgt, da jeg føler mig meget mere tilpas i C# kode, fremfor XAML kode. Derfor bliver alle tekstbox, datagridden, og deres værdier sat i C# kode. Jeg fandt også ud af for sent, at jeg ikke havde det særlig godt med binding, og derfor er der mangel på dette i opgaven. Så det skal jeg have læst op til, til eksamensprojektet.

Jeg valgte datagrid til at fremvise listen af udlejninger, da jeg føler den er meget passende i forhold til at der kan sorteres efter, hvad den enkelte bruger ønsker den sorteret efter. Alt i alt har jeg valgt datagrid pga. dens mange automatiseret egenskaber, såsom kolonnenavne, der oprettes ud fra de valgte elementer.

Jeg har kodet søgningsfeltet(Der du indtaster email), således at hvis databasen har en email som hedder ”hans@gmail.com”, så vil du kunne finde de udlejninger der er koblet op på den email selvom du kom til at skrive hans med stort H. Dette har jeg valgt at gøre, da det er en væsentlig lille fejl, en bruger ikke bør tænke på. Dette har jeg løst ved at når du søger på en email bliver alle bogstaver gjort til små bogstaver, og derudover når der oprettes en kunde i web-delen, så vil kundens email, uanset indtastning, gemmes i databasen som små bogstaver.

I forhold til status delen og hvordan den blev løst. Så vaklede jeg mellem om jeg skulle benytte mig af ENUM, men da det var meget præcist, hvad der skulle stå i status feltet, valgte jeg at brugeren kan ændre det ved hjælp af en drop down box, hvori jeg har hardcoded de 3 forskellige værdier en status kan have, og derfor er status valgt til blot at være en String, både i databasen og i model klassen. Jeg så ikke fordelen i lige netop denne situation for at lave en enum med de værdier, når den meget simpelt kunne laves på den anden måde.

I min model klasse har jeg valgt mine datoer, som en datetime, dette fandt jeg senere ud af måske ikke var så hensigtsmæssig, da der ikke kræves et tidspunkt på udlejningen. Jeg opdagede fejlen, så sent, at det ville koste for meget tid at lave det om. Så derfor vil der når der oprettes en udlejning laves en default værdi for tidspunktet på datetime.

Selve WPF delen er lavet meget simpelt, på den måde at det hele foregår på en side. Dette er gjort fordi der ikke var noget krav angående flere sider, samt fordi at alt data kunne være på en side uden at det blev uoverskuelig.

Jeg har dog tilføjet et par messagebox, ved enden fejl meddelelser eller når man vil slette en udlejning.

# Web:

I web delen, er selve designet er lavet meget efter Entity Framework.

I koden startede jeg med at sende parameter med ved hjælp af url’en. Men fordi jeg senere endte med ikke kunne få løst at få det sendt hele vejen med til oprettelsessiden for udlejning, så valgte jeg senere at prøve at begå mig med Session, hvilket løste problemet. Derfor finder du i denne opgave både eksempler hvor data er sendt videre via url, og samtidig hvor dataen er sendt videre via Session.

Kunden i Session bliver sat allerede når du logger ind, og dette kan selvfølgelig skabe problemer ved at ændre i url’en efter du har logget ind, da det stadig ville være den samme kunde som er gemt, indtil du logger ind med en ny kunde. Da opgaven siger vi ikke behøver at tænke sikkerhed, har jeg ikke løst dette problem, da det ville være omfattende.

Alle nye kunder oprettes med en email hvor alle bogstaver er små, men samtidig kan du, som i WPF-delen, godt logge på selvom du skriver din email med stort. Hvis email’en som indtastes ved login findes i databasen, så vil den tjekke om password passer med email’en, stemmer de ikke overens, så vil den reload login siden med tomme felter. Derimod hvis email’en ikke findes i databasen oprettes en ny kunde med den valgte email og password.

Jeg valgte også at lave session løsning på accepter/annuller udlejningsdelen. Der stødte jeg ind på et problem hvor at når jeg sendte udlejningen videre via session til bekræftelsessiden, så oprettede den et nyt identisk værktøj i databasen, som der var valgt til udlejningen, det vil sige at jeg pludselig sad med mange af samme værktøj. Dette løste jeg ved at man ved oprettelsen og saveChanges() satte den virtuelle parameter Værktøj på udlejningen(sendt videre via session) til null, da det var den jeg tror, udløste problemet. Det løste det i hvert fald for mig.

Ellers er meget af koden jo autogenereret ud fra entity framework.

# Database og SQL:

For oprettelse af databasen, tabeller og testdata, kommer der i mit bilag til at ligge sql koden til det, samt jeg vil også ligge det i en folder på min solution i visual studio.