

**Wiki****[Dokumentation](#) • [Database](#) • [C#](#) • [Tests](#) • [Log, Tidsplan](#) • [Manual](#)**

## Log og Tidsplan

| TIDSPLAN       | LOG            |
|----------------|----------------|
| Hvad er planen | Hvad skete der |

### 9/3

#### Log

Udlevering af opgaven.

Udarbejdning af ER-Diagram, via ERD-editor. De grundlæggende tabeller er blevet lavet, og vi arbejder med at finde ud af hvilket relationer der refererer til hvad. ERD editor har også givet os variabler til C# objekterne der skal oprettes, så i consol applikationen mangler der umiddelbart brugerinterface, og andre en hel masse andre ting

MS SQL server er oprettet som en ekstern virtuel maskine, kan tilgås online. Use MS SQL connect: [w2k19sql.hq.gollomotors.dk](http://w2k19sql.hq.gollomotors.dk)  
Obs: Der skal fortsat rettes lidt i rettighederne for denne bruger!!

### 10/3

#### Plan

ER - diagram redigering redigeres fortsat:

- specificering af relationer
- specificering af datatyper

#### Log

-Database er oprettet, med et udkast af de forskellige elementer der skal bruges -lavet tomme klasser

Ting der skal laves -en application der kan vise forskellige elementer af databasen, alt efter hvilket rettigheder de enkelte brugere skal have -index -udarbejde de forskellige slags kontrakter -potentielt lave et kalenderprogram så de enkelte datetimer værdier ikke modstrider hinanden -Udarbejde algoritme for prissætning af udlejning og diverse betalinger. og oprette pris i databasen -Et modul til at oprette queries -Implementering af hvorvidt en konkret feriebolig er udlejet eller ej, sker i 2 tempi: I objektet der repræsenterer en feriebolig haves et multidimensionelt array med x = årstal, y = 52 uger og z = en int der er 0 hvis kahytten er ledig, og ellers har et udlejningskontrakt id. I databasen repræsenteres det samme, men det skal vikles ind og ud til objekterne til og fra databasen for at kan bruge eller gemme ændringer..... Basalt gemmes hvilke uge nr og årstal der er udlejet for en given

ferieboligs id. Der gemmes også et udlejnings kontrakt nr. (selvom det er redundant) og hele møllen skal sorteres før det fyldes ind i et objekt. Når det skal indsættes i denne tabel fyldes de samme data blot ind i røven af tabellen Dvs. at de konkret udlejningsdata skal graves ud via et udlejnings kontrakt id.

Denne analyse er konkluderet fordi det er sært gemme lister i en database, men det er ret nemt at navigere i og repræsentere multidimensionelt arrays

Efter ER-diagrammet er nogen lunde på plads, ligner det at de to udlejningsformer kan repræsenteres i en database, men med en divergerende OOP-struktur der dog bygger på noget arv eller indlejring eller genetisk: en feriebolig, sommerhus eller lejlighed, er nogenlunde ens. Den store forskel er at en lejlighed ligger inde i et lejlighedskompleks. Lejlighedskomplekset kan derfor godt have en anden og overordnet type administrator (1 til mange) end en feriebolig. Begge typer administrator er iforvejen en til mange... Tænke tænke....

## 11/3

### Plan

- Lave views og indexes til DB:

programmet og vi skal vide hvilke opslag der skal skal optimeres i databasen, hvad der spørges på.

- OoP synk:

Vi skal have et overblik over hvad hvile klasser skal gøre og hvorfor.  
En forståelse over hvordan databasen virker og gderigennem:

- Kortlægning af hvad programmet skal kunde:

1. Hvad kan databasen gøre for os, svarer på direkte
2. Klasser der kan håndter programmet på; Måder vi vil have det til at virke

på. f.eks.:

1. for at kunne udskrive en kontrakt.
2. Se hvilke huse der er ledige i et givent distrikt i en given uge.
3. Gui
4. kortlægning af en eller anden fornuftig grænse flade
5. Krav der står i opgaven:
6. fjern unødige wonder kalsser fra ovenstående design, i første omgang:

### Log

## 12/3

### Plan

1. Domumentation
2. koden klasser der kan æde og fodre database kød og har metoder til en eller anden fornuftig grænse flade.

### Log

- ✓ Dokumentation er klargjort til at blive taget i brug på:  
<http://wiki.w2k19sql.hq.gollomotors.dk/>
- ✓ Databasen er etableret i en reduceret form vha script.  
sydvest-bo\_Create\_DB\_export\_from\_ERD\_Editor.sql  
Scriptet sletter databasen og genopretter den, samt giver vores tec  
bruger sine rettigheder tilbage igen efter slætning.

## 13/3 Fredag

### Plan

1. En eller anden opgave fordeling til weekenden.

### Log

- ½ er i gang med at lave et script der kan initialisere databasen med nogle default værdier via et c# projekt der ligger i mappen:  
\_init\_empty\_database  
programmet vil have database passwordet som argument:  
Højreklik på \_init\_empty\_database i højre side  
-> properties -> debug -> arguments  
OBS: Husk at slette disse inden der synkes med github.
- ½ Projektet har også de aktuelle klasser der matcher databasen i en classe file. men der mangler stadig constructorer.
- ✓ programmet afprøver også en metode til at maximere consolvinduet
- ✓ Der er lavet et test projekt til at finde max console windows size og som har en scrolle feature der anvender screenbuffer så man kan få vist mere en skærmen kan trække:  
Console\_Screen\_max\_test2\_bufferscrole

## 14/3weekend

### Plan

### Log

## 15/3weekend

### Plan

### Log

## 16/3

### Plan

1. Domumentation
2. program opfølgning

3. konstiens check
4. gui

## Log

- Foreløbige Sydvest-Bo\_test\_DATAFILER.csv er lavet og ligger i mappen \Resourser  
Database, 6. Test Data

## 17/3

### Plan

1. gui
2. Domumentation
3. Lir:
  - a. Gendindførsel af hidtidige unødige og fjernede wonder klasser, der

tidligere blev fjernet fra ovenstående design,:

1. DokuWiki

## Log

## 18/3 Deadline

### Plan

1. Lir
2. Gui
3. Aflevering af opgave kl :
4. øve fremlæg.

## Log

## 19-20/3

### Plan

## Log

1. Fremlæg (corona)

## Ekstra

✓ DokuWiki - Det hedder faget jo email afsending - Skal programmet virke eller ej

