

## Tur&Retur kørselslogbog (simpel udgave)

Før du starter på denne opgave skal du have udført:

- Intro til **C#**
- Intro til **UML** og **OOAD**
- Introduktion til databaser
- Mini guide til SQL

### Introduktion:

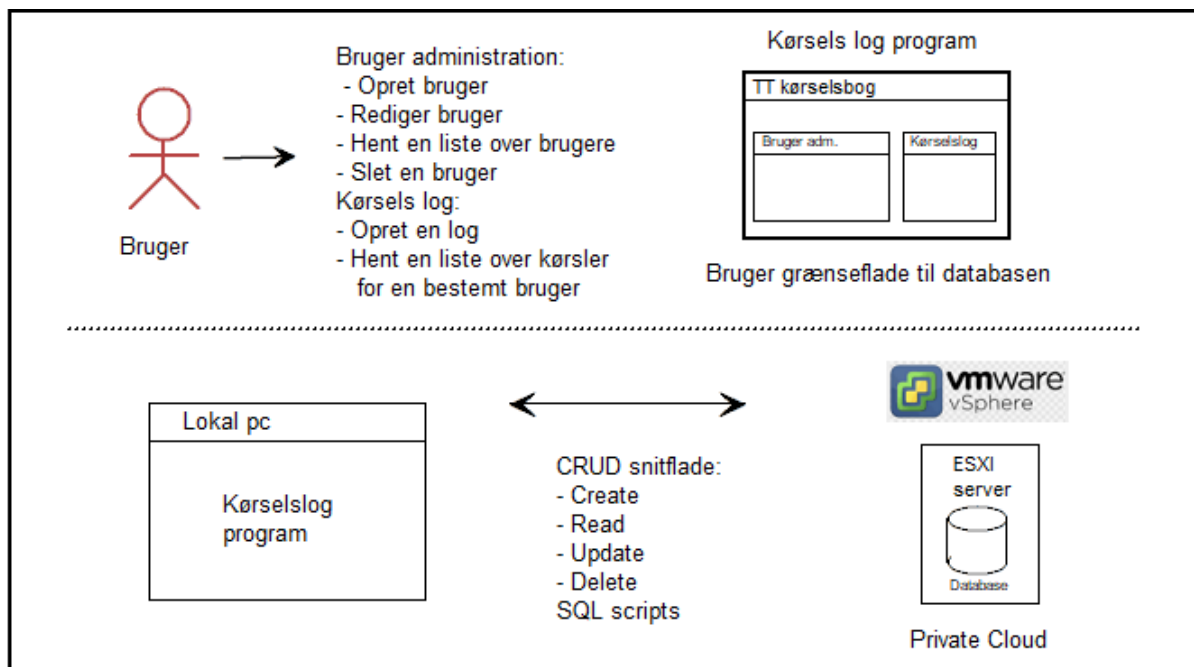
Praktikcenteret har fået en opgave af transportfirmaet Tur&Retur. De skal bruge et kørselslogprogram, der holder styr på hvor mange kilometer deres biler kører pr. opgave.

Anvend den viste brugergrænsefladen i nedenstående mockup til opgaven. De 4 forklarende tekster skal ikke med i mockuppen.

Opgaven består af flere trin:

1. Afklaring og afgrænsning af opgaven.
2. Lav en overordnet løsningsmodel (vis hvordan opgaven kan løses i **praksis** i en **tegning**)  
Se nedenstående billede.
3. Udarbejd et Work Break Down skema **WBS** for projektet med de opgaver der fremgår af dels mockuppen, se side 3 og dels løsningsmodellen. På side 4 er der et eksempel på et Gantt-Diagram.
4. Krav til programmet, se side 3.

2. Overordnet løsnings model: (Simplificeret model af opgaven som den kunne være tænkt løst.)



### Opgaver med brugerinteraktion:

Med udgangspunkt de handlinger som brugeren kan foretage sig skal der laves en liste med klasser og dertilhørende metoder.

### Opgave med Brugergænsefladen:

- Afklaring af brugerens behov / krav / adgang til data.
- Fremstilling af grænsefladen.
- Forventningsafstemning til grænsefladen

### Opgaver med SQL serveren:

- E/R diagram
- DB diagram
- Snitflade (CRUD)

Dato: 16/11 – 2020 - KMJ

### 3. WBS:

Projekt: Kørselslogbog	Startdato:	Slutdato	Status:	Resource:
1.0 OOAD				
1.1 Analyse				
1.1 Use cases + flowcharts				
1.1.1 Opret bruger				
1.1.2 Rediger bruger				
1.1.3 Slet bruger				
1.1.4 Liste over brugere				
1.1.5 Opret en kørselslog				
1.1.5 Liste over kørselslogs				
1.2. Design (C# og SQL)				
1.2.1 Database				
1.2.1.1 E/R diagram				
1.2.1.2 DB diagram				
1.2.1.3 SQL scripts				
1.2.2 Klasser				
1.2.2.1 Klasse diagram				
2.0 OOP				
2.1 Programmering af win form klassen				

Til hver opgave i WBS'en i punkt 3 skal der laves en beskrivelse der indeholder de nødvendige informationer til løsning af opgaven.

### OOP: (programmering)

Projekt navn:	Tur&Retur kørselslogbog
Klasse navn:	DataIO
Metode navn:	CreateUser
Parametre (type, text):	String Navn, String NummerPlade
Retur værdi type:	Boolean
Opgave beskrivelse:	Hent de indtastede data Check data for ulovlige tegn. Opret en bruger i databasen Returner status på database operationen
Fejl håndtering:	Udskriv en fejl besked til brugeren se følgende: Databasen er ikke tilgængelig Der er ulovlig tegn i data input Bruger er ikke blevet oprettet

## 1. Mockup til kørsels log applikationen.

The mockup is titled 'Kørsels log app' and is divided into two main sections: 'Personale data' and 'Kørsels log registrering'.

**Personale data:**

- Opret stamdata:** Fields for Navn, Dato, and Nr. plade. Buttons: Cancel, Ok.
- Rediger stamdata:** Fields for Navn (with a dropdown arrow), Dato, and Nr. plade. Buttons: Cancel, Ok.
- Slet stamdata:** Field for Navn (with a dropdown arrow). Buttons: Cancel, Ok.

**Opret en bruger:** Indtast et navn og en nummer plade. Klik på **Ok** for at oprette en record og få det vist i nedenstående liste. Ved klik på **Cancel** resettes felterne.

**Rediger en bruger:** Vælg en bruger fra drop down listen. Rediger nummer plade teksten. Klik på **Ok** for at gemme den opdaterede record of det vist i nedenstående liste. Ved klik på **Cancel** resettes felterne.

**Slet en bruger:** Vælg en bruger fra drop down listen. Klik på **Ok** for at slette den bruger (record) fra den nedenstående liste. Ved klik på **Cancel** resettes navne feltet.

Navn:	Dato:	Nummer plade:

**Kørsels log registrering:**

- Opret kørsels log :** Fields for Navn (with a dropdown arrow), Dato, Nr. plade, and Opgave. Buttons: Cancel, Ok.

**Registrer en kørsel:** Vælg en bruger fra drop down listen. Indtast en kort opgave tekst. Klik på **Ok** for at gemme den opdaterede record of det vist i nedenstående liste. Ved klik på **Cancel** resettes felterne.

Navn:	Dato:	Nr. plade:	Opgave:

De opgave der kan findes i den overordnede løsning / mockuppen skal skrives ind i WBS'en. Hvis en opgaven er for stor skal den opdeles yderligt i flere mindr opgaver.

### Krav:

Til opgaven skal der anvendes følgende:

#### OOAD:

- Use case + specification + flowchart for hver enkelt handling
- Klasser med metoder og attributter = klassediagram med angivelse af arv, relation(er), aggregat(er)
- Databasesdesign – E/R diagram, DB diagram, SQL scripts, SQL forespørgelser

#### OOP:

- Anvend de fundne klasser i klassediagrammet til fremstilling af en Win form klasse med attributter og metoder med tilhørende parametre

Der anvendes en virtuel server i en **client/server** opsætning.

Den virtuelle server skal indholde en Win server 2019 og en MSSQL server 2019.

Programmet skal have et UI på dansk.

Programmet skal laves i Windows Forms.

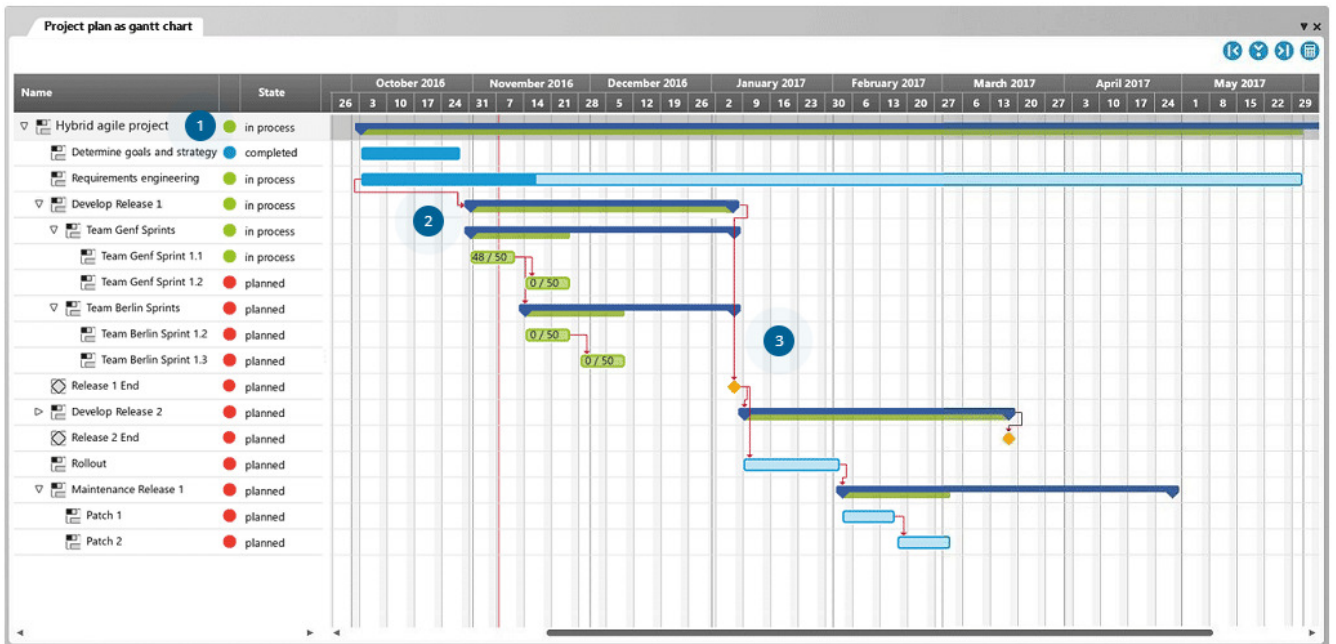
Databasen skal kunne oprettes vha. et SQL script (query) i SQL Server Management Studio SSMS (husk at lave en backup af db).

Se vedlagte beskrivelse af opsætning af Windows 2016/19 server på EXSI Vmware virtuel server.

Al kode (C# og SQL scripts) skal dokumenteres og lægges som bilag.

Dato: 16/11 – 2020 - KMJ

Eksempel på et Gannt diagram:



Det vist Gannt diagram består af en WBS struktur og en visuel planlægnings kalender.