

# Delfin svømmeklub

<b>Use case</b>	<b>2</b>
<b>Use case diagram</b>	<b>3</b>
<b>SSD UC# 1</b>	<b>6</b>
<b>Domæne Model:</b>	<b>8</b>
<b>Class model:</b>	<b>9</b>
<b>SWOT</b>	<b>11</b>
<b>Interessentanalyse</b>	<b>13</b>
<b>Gantt kort</b>	<b>15</b>

## Use case

Svømmeklubben Delfinen **er** en mindre **klub**, der **er** i vækst. Klubbens **ledelse** **ønsker** derfor **udviklet** et administrativt **system** til at **styre** **medlemsoplysninger**, **kontingenter** og **svømmeresultater**.

Det **er** klubbens **formand**, der **tager** sig af nye **medlemmer**. Ved indmeldelse i **klubben** **registreres** diverse **stamoplysninger** om **personen** herunder **alder**. Desuden **registreres** **oplysninger** om personens **ønskede aktivitetsform**, det **vil** sige aktivt eller passivt **medlemskab**, **junior** eller **senior svømmer**, **motionist** eller **konkurrencesvømmer**.

Klubbens **kasserer** **tager** sig af alt vedrørende **kontingentbetaling**. **Kontingentets** størrelse **er** betinget af flere **forhold**. For aktive **medlemmer** **er** **kontingentet** for **ungdomssvømmere** (under 18 **år**) 1000 årligt, for **seniorsvømmere** (18 **år** og over) 1600 kr. årligt. For **medlemmer** over 60 **år** **gives** der 25 % **rabat** af **senior taksten**. For passivt **medlemskab** **er** **taksten** 500 kr. årligt.

**Kassereren** **har** ønsket, at systemet **kan vise** en oversigt over **medlemmer**, der **er** i restance.

**Konkurrencesvømmerne** **har** tilknyttet en **træner**. **Konkurrencesvømmerne** **er** inddelt i 2 **hold** efter **alder**. **Ungdomsholdet** **er** for **svømmere** under 18 **år**. **Seniorholdet** **er** for **svømmere** på 18 og over. Hver **konkurrencesvømmer** **er** desuden registreret i forhold til hvilke **svømmediscipliner**, han **er** aktiv i.

Inden for hver **svømmedisciplin** **registreres** den enkelte **svømmers** bedste **træningsresultat** og **dato løbende**. For de **svømmere**, der **har deltaget** i **konkurrencer**, **registreres stævne**, **placering** og **tid**. Det **er** på **baggrund** af de enkelte **svømmers resultater**, at **træneren** udtager **svømmere** til deltagelse i **konkurrencer**. **Træneren** **ønsker** derfor en **oversigt**, der **kan vise** klubbens top 5 **svømmere** inden for hver **svømmedisciplin**.

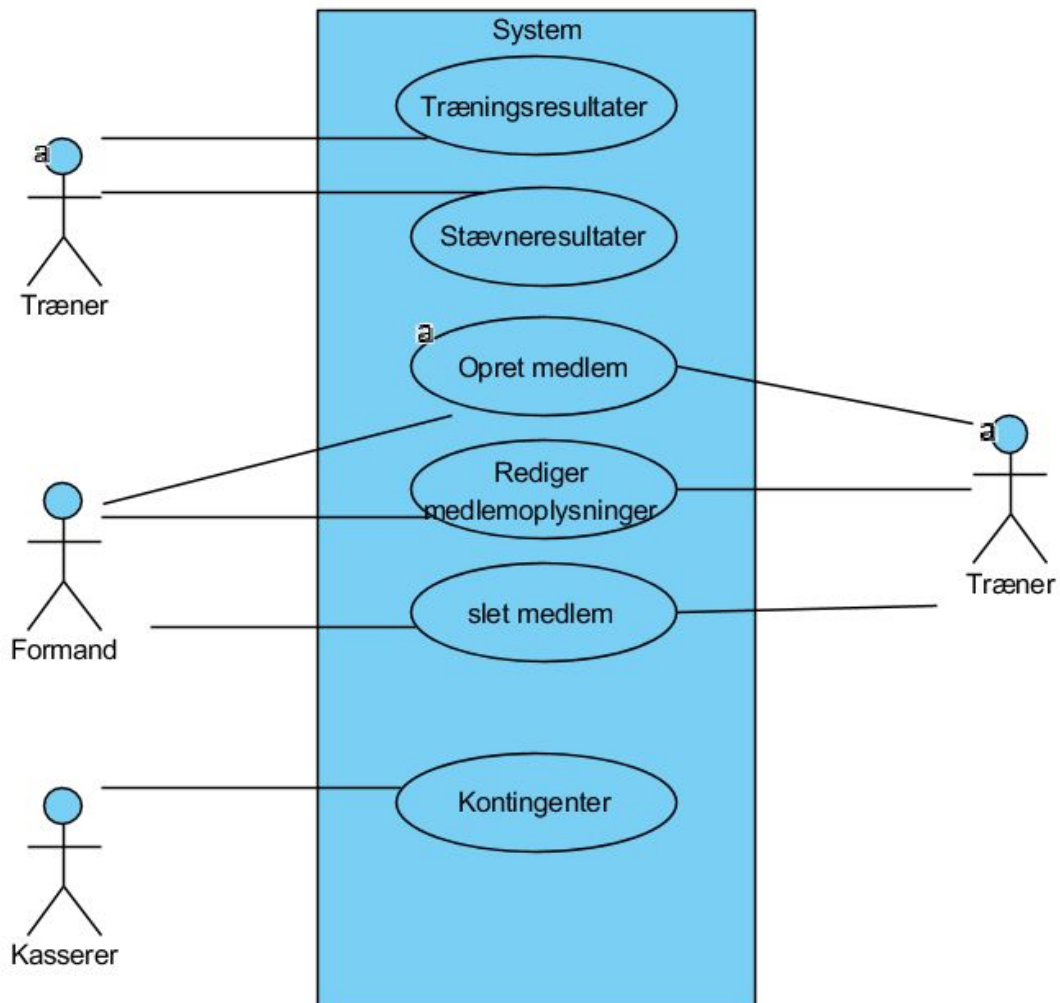
### Navneord/Noun:

Svømmeklubben, **klub**, **ledelse**, **system**, **medlemsoplysninger**, **kontingenter**, **svømmeresultater**, **klubbens**, **formand**, **medlemmer**, **stamoplysninger**, **personen**, **alder**, **oplysninger**, **aktivitetsform**, **medlemskab**, **junior**, **senior**, **svømmer**, **motionist**, **kasserer**, **kontingentbetaling**, **forhold**, **ungdomssvømmere**, **seniorsvømmere**, **taksten**, **svømmedisciplin**, **svømmere**, **resultater**, **klubbens top**, **oversigt**, **baggrund**, **tid**, **placering**, **stævne**, **konkurrencer**, **dato**, **træningsresultat**, **svømmere**, **seniorholdet**, **træner**, **konkurrencesvømmer**, **ungdomsholdet**, **hold**, **senior taksten**, **rabat**, **år**, **Kassereren**.

### Udsagnsord/Verbs:

**Er**, **ønsker**, **udviklet**, **styrer**, **tager**, **registreres**, **ønskede**, **vil**, **gives**, **har**, **kan**, **vise**, **løbende**, **deltaget**

## Use case diagram



Use case diagrammet viser sammenspil mellem actors og use casene.  
En use case bruges til at analysere systemets krav.

UC# 1 er en fully dressed use case, der eksisterer 3 forskellige udgaver af use cases breif, casual og fully dressed. Fully dressed er den mest uddybende udgave af disse 3.

**UC# 1 Træningsresultater:**

**Scope:** Resultats system, oplysninger om resultater fra træning

**Level:** At se, oprette, redigere og slette træningsresultater

**Primary actor:** Træner

**Stakeholders and Interests:**

Træneren: har brug for et system til at holde styr på svømmernes træningsresultater.

Svømmerne: vil have et system som holder styr på deres træningsresultater.

**Preconditions:** Træner har brug for at kunne se, oprette, redigere og slette træningsresultater.

**Postconditions (success guarantee):** Resultaterne bliver set, oprettet, redigeret eller slettet.

**Main success scenario:**

1. Træneren logger ind i systemet
2. Træneren vælger "Træningsresultater"
3. Træneren vælger "Se træningsresultater"
4. Træneren logger ud

**Alternative flow:**

1. Træneren skal oprette et Træningsresultat
  - Træneren vælger "Træningsresultater"
  - Træneren vælger "Opret Træningsresultater"
  - Træneren vælger "Gem Træningsresultater"
  - Træneren logger ud
2. Træneren redigere Træningsresultat
  - Træneren vælger "Træningsresultater"
  - Træneren vælger "Rediger Træningsresultater"
  - Træneren vælger "Gem Træningsresultater"
  - Træneren logger ud
3. Træneren sletter et Træningsresultat
  - Træneren vælger "Træningsresultater"
  - Træneren vælger "Rediger Træningsresultater"
  - Træneren vælger "Slet Træningsresultater"
  - Systemet prompter "Er du sikker på at du vil slette"
  - Træneren vælger ja
  - Træneren logger ud
4. Indtaster forkert kode
  - Træneren logger ind med forkert kode
  - System skriver forkert kode prøv igen

**Special requirements:** Komplet computer

**Frequency of occurrence:**

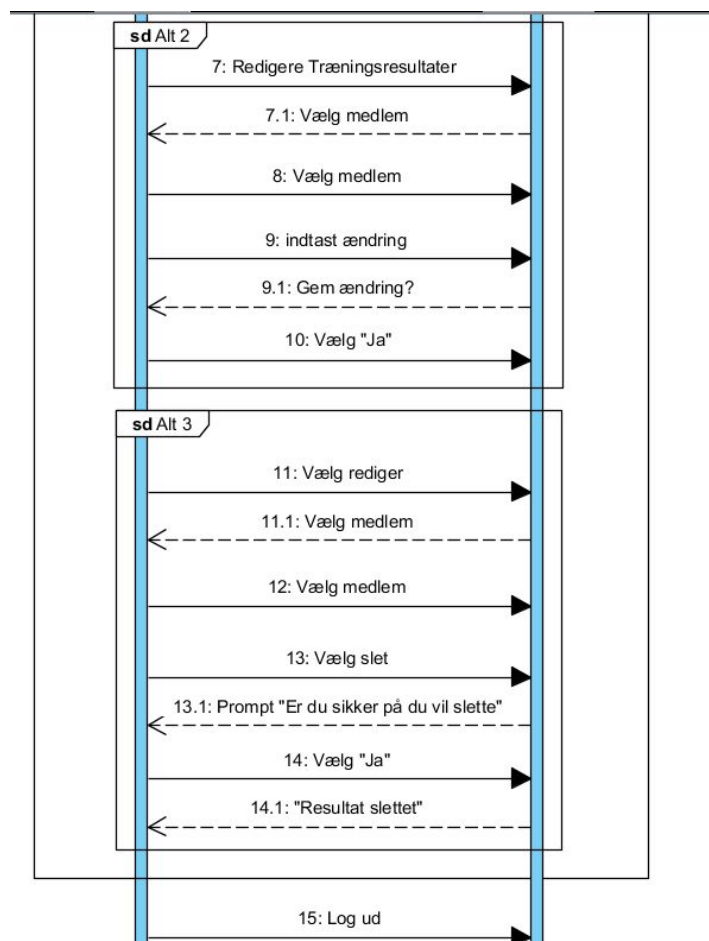
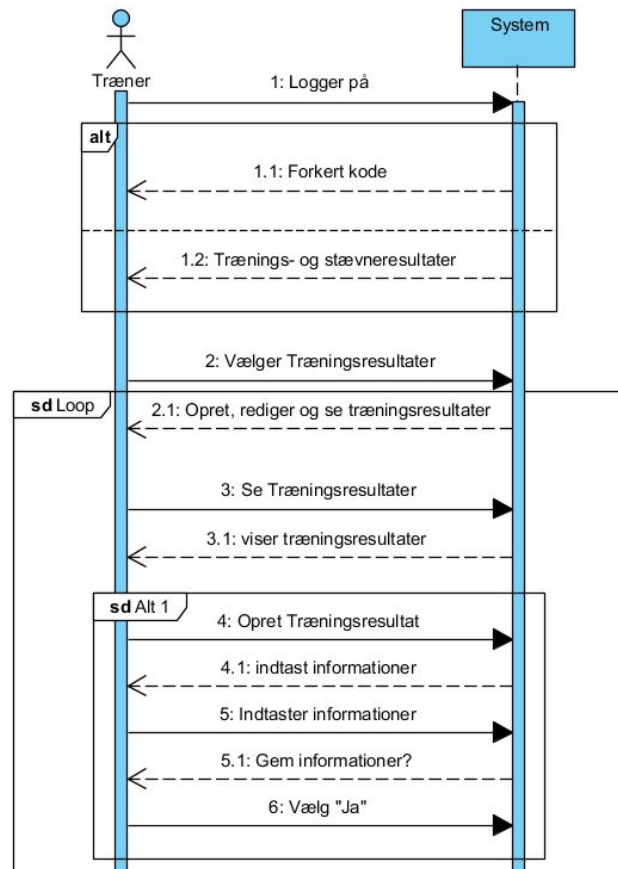
Når træneren har brug for at se eller ændre resultater.

Dette kan forekomme op til flere gange dagligt.

## SSD UC# 1

SSD viser input fra actor og output fra systemet et skridt af gangen. Der er mulighed for at vise når der er et loop mm.

Vores SSD viser input fra træneren angående træningsresultater og tilhørende output fra systemet.



**UC# 2 stævneresultater:**

**Primary actor:** Træner

**Main success scenario:**

1. Træneren logger ind i systemet
2. Træneren vælger "Stævneresultater"
3. Træneren vælger "Se/opret/rediger Stævneresultater"
4. Træneren logger ud

**UC# 3 Opret medlem:**

**Primary actor:** Formand

**Main success scenario:**

1. Formanden logger ind i systemet
2. Formanden vælger "opret medlem"
3. Formanden indtaster nye medlems information
4. Formanden logger ud

**UC# 4 rediger medlemsoplysninger:**

**Primary actor:** Formand

**Main success scenario:**

1. Formanden logger ind i systemet
2. Formanden vælger "rediger medlemsoplysninger"
3. Formanden indtaster medlemmets ændringer
4. Formanden logger ud

**UC# 5 slet medlem:**

**Primary actor:** Formand

**Main success scenario:**

1. Formanden logger ind i systemet
2. Formanden vælger "rediger medlemsoplysninger"
3. Formanden vælger "slet medlem"
4. Systemet prompter "Er du sikker på at du vil slette medlem?"
5. Formanden vælger "Ja"
6. Formanden logger ud

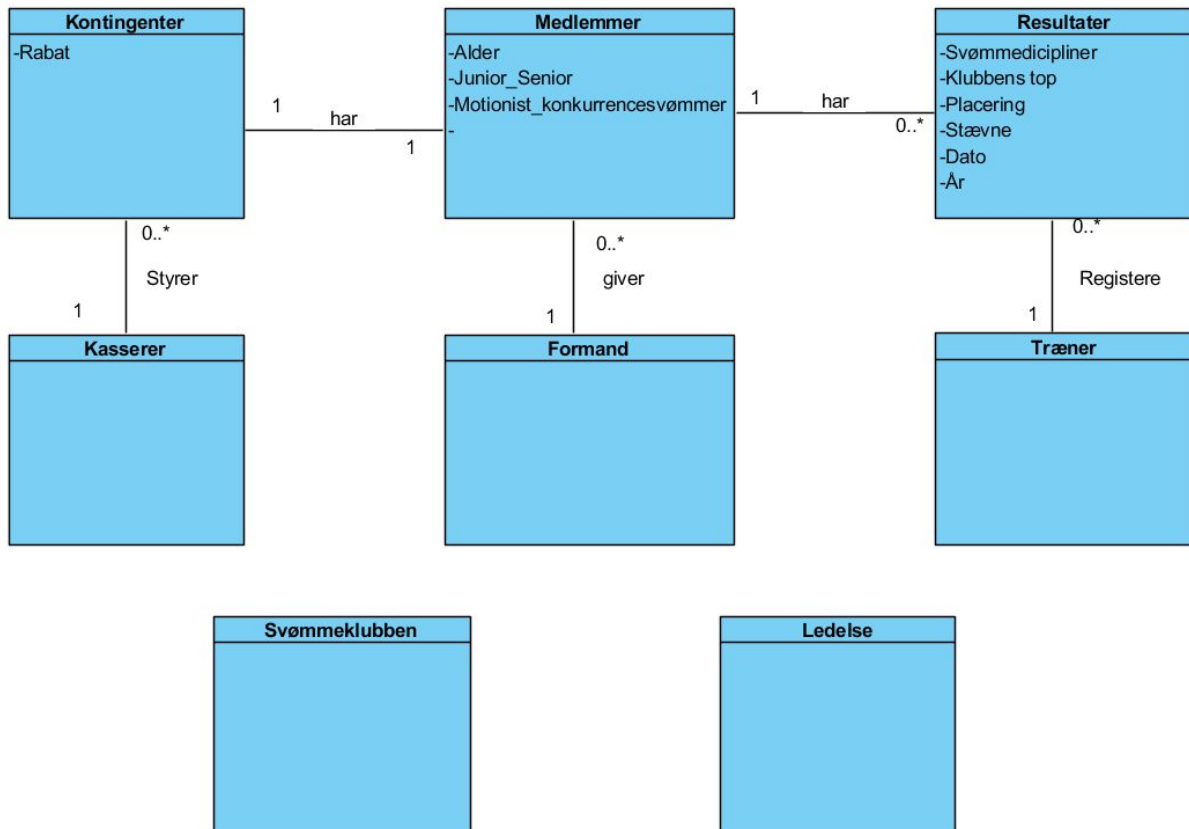
**UC# 6 kontingent:**

**Primary actor:** kasser

**Main success scenario:**

1. Kassereren logger ind i systemet
2. Kassereren vælger "kontingent"
3. Kassereren opdatere/tjekker kontingent statuser
4. Kassereren logger ud

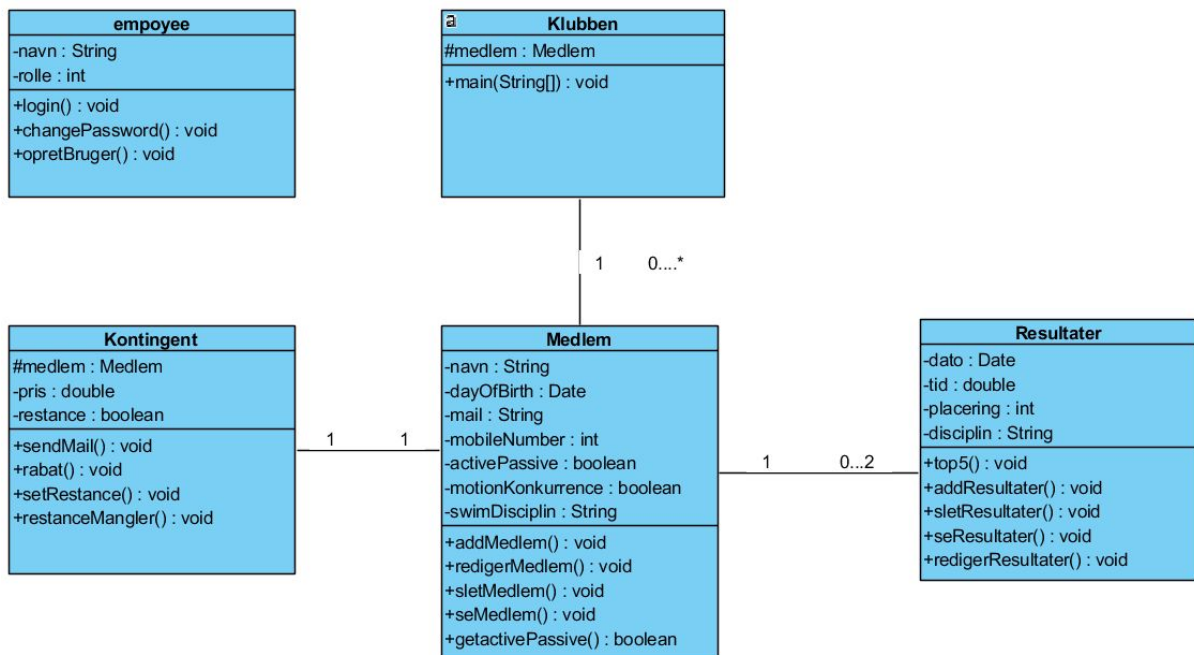
## Domæne Model:



Domæne modellen bliver lavet ved hjælp af navne- og udsagnsord vi har fået fra use casene hvor navneordene bliver til klasser og attributter og udsagnsordene bliver til metoder som binder klasserne sammen.

Ud fra vores navne og udsagnsord er vi kommet frem til domæne modellen som kan ses ovenfor.

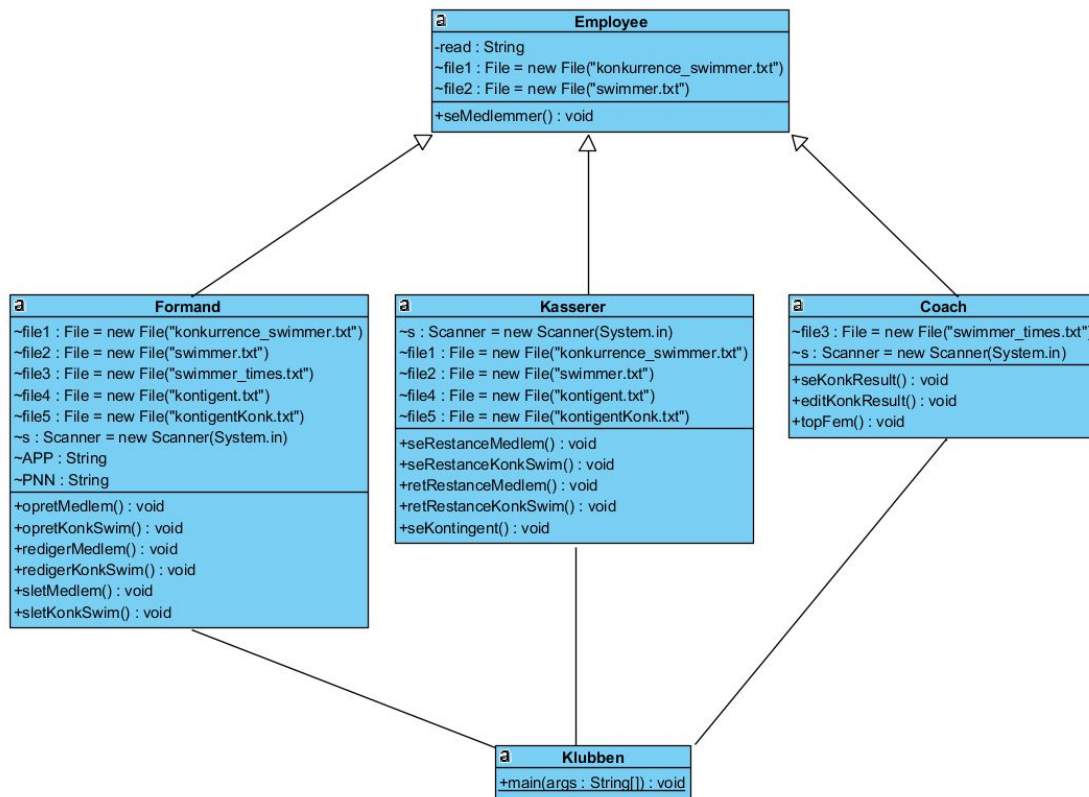
## Klasse diagram:



Første udgave af vores klasse diagram, som viser metoder og attributter i de forskellige klasser og sammenhæng mellem klasserne.



## 2nd udkast



En opdateret version som viser hvordan programmet er sammensat.

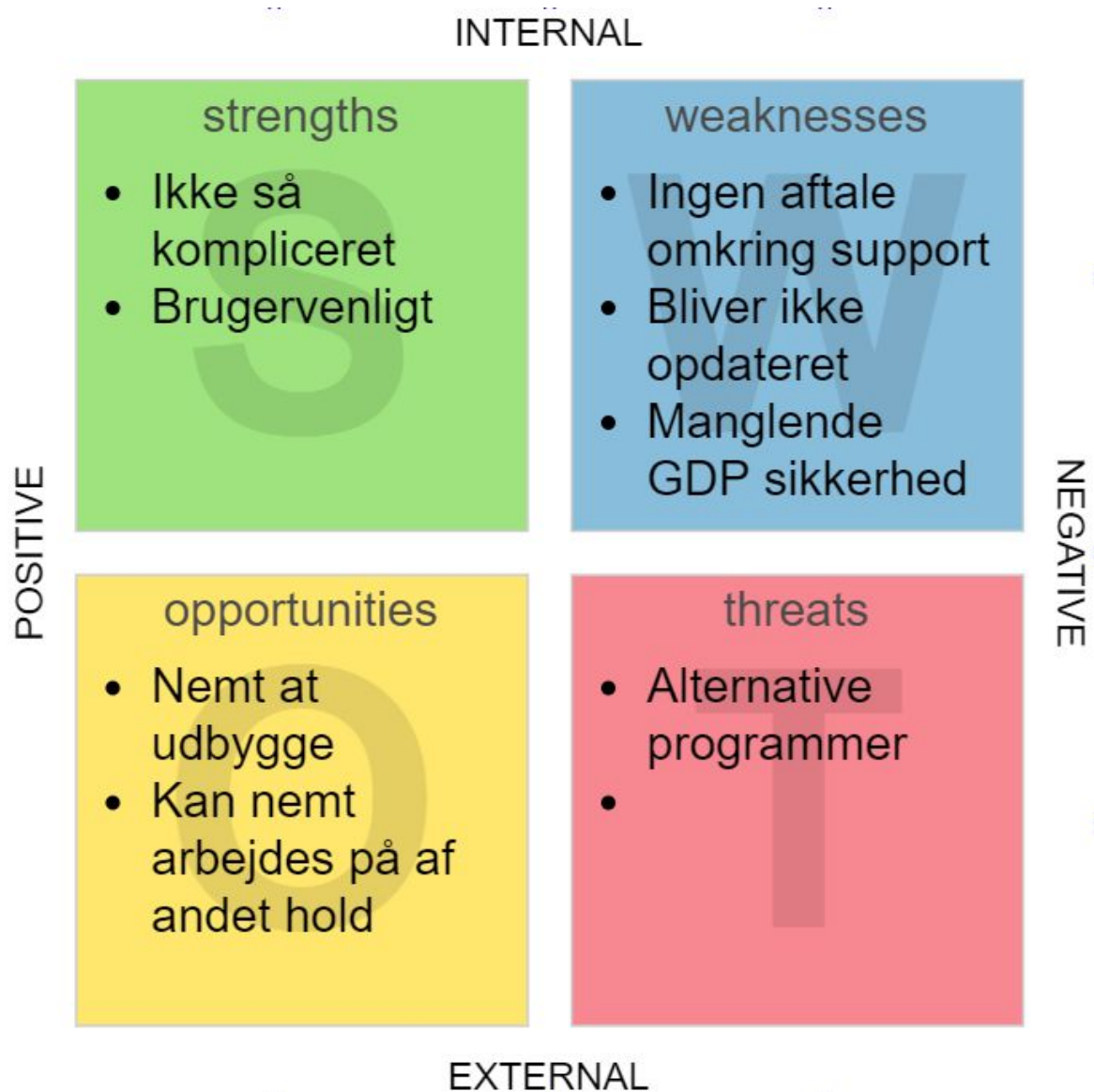
Som det kan ses i diagrammet arver Formand, Kasserer og Coach fra employee klassen.

Vores main ligger i klassen "Klubben" og laver objekter ud af klasserne Formand, Kasserer og Coach.

Den store forskel fra vores første udkast af klasse diagrammet er at klasserne i den første blev til de dokumenter vi gemmer information i. Vi valgte at der var forskel på hvad formand, kasserer og træner skulle have adgang til og derfor gav det mere mening at opdele metoderne efter den person som skulle have adgang til dem.

Vi valgte at lade formand, kasserer og træner arve fra employee klassen da de alle skulle have denne metode.

# SWOT



En analyse over styrker, svagheder, muligheder og trusler, som kan bruges til at undersøge hvordan en "virksomhed/gruppe" kan opnå bedre/større resultater og være opmærksom på hvilke problem man har/kan få.

S: Vores program arbejder med simple valg, så man nemt kan finde rundt i systemet.

Hver person har kun adgang til de funktion som de skal bruge.

W: Der er ikke nogle aftaler om videreudvikling af programmet eller support hvis der skulle opstå fejl

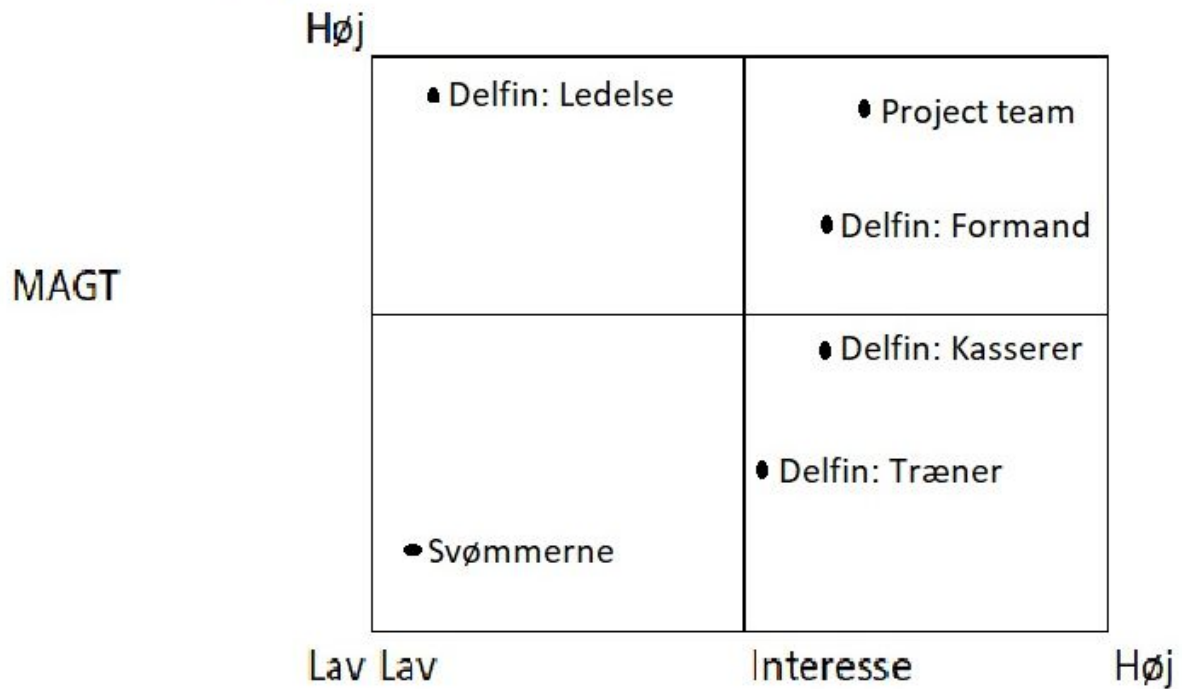
Programmet er ikke designet efter GDP fx bliver personfølsomme data i et simpelt text dokument (ubeskyttet).

O: Det er nemt at udbygge eller opdatere funktioner i programmet, og der er indlagt kommentare til koden så man nemt kan se hvad de forskellige funktion gør

T: Da det er et simpelt program kan det nemt erstattes (eventuelt af excel).

## Interessentanalyse

### Interessentanalyse



Viser hvor meget magt og interesse de forskellige parter har i systemet. Det kan bruges til at se hvem der er beslutningstager og hvem der skal bruge det og derfor har forslag til designet

## Risikoanalyse

Risikomoment ▲	Sandsynlighed	Konsekvens	Produkt	Præventive tiltag	Ansvarlig	Løsnings forslag	Ansvarlig 2
Defekt udstyr	3	4	12	Alle project medlemmer medbringer computere	Gruppe 5	Project ligger på GIT hub	Gruppe 5
Forsinkelse i levering	1	10	10	Holde øje med Gantt og sørger for projectet er på tidsplan	Mads	Løbende kontakt med Delfin	Mads
Mangelfulde testninger	2	8	16	Få udenfra kommen til at teste programmet	Nikas	Crunch time	Nikas
Nøgleperson forlader projectet	2	7	14	Godt arbejdsmiljø i gruppen, ingen arbejder alene	Gruppe 5	Holde hinanden opdateret løbende	Gruppe 5
Problemer med installation	3	4	12	Prøve at installerer det på anden maskine med samme OS	Jesper	Aftale med anden gruppe omkring support	Jesper
Sygdom	5	3	15	Spise vitaminer, sundt og ordenlig beklædning	Gruppe 5	Informere gruppen, arbejd hjemme	Gruppe 5
Ændringer til krav i systemtet	5	3	15	Løbende kontakt med Delfin og god programmerings praksis	Gruppe 5	Deadline for sidste ændringer.	Gruppe 5

Risikoanalysen er et forberedende værktøj som kan bruges hvis nogle af de listede risicier hænder. I risikoanalysen ligger løsningsforslag og præventive tiltag.

# Gantt kort

## Delfin Gantt Kort



Gantt kortet er et planlægningsværktøj, så man kan holde styr på hvem der laver hvad hvornår. Gantt kortet bliver selvfølgelig opdateret løbende så man kan holde styr på tidsplanen og se om man er on track.