

Svømmeklubben Delfinen

Datamatikeruddannelsen  
DAT 17C

1. semester rapport

08-12-2017

Gruppemedlemmer:

|  |  |
| --- | --- |
| Casper Frost Andersen | 14-02-90 |
| Martin Løseth Jensen | 12-04-97 |
| Christian Strunge | 18-11-92 |
| Rasmus Sadurski | 23-12-87 |

Indhold

[Indledning 3](#_Toc500234050)

[Problemformulering 3](#_Toc500234051)

[Afgrænsning 4](#_Toc500234052)

[Interessent analyse 4](#_Toc500234053)

[Brainstorm 4](#_Toc500234054)

[Sortering 5](#_Toc500234055)

[Svømmeklubbens matrix 5](#_Toc500234056)

[Systemudviklernes matrix 5](#_Toc500234057)

[Interessenterne 6](#_Toc500234058)

[Integrer 6](#_Toc500234059)

[Interessent matrix 8](#_Toc500234060)

[SWOT-analyse 9](#_Toc500234061)

[Set ud fra svømmeklubben 9](#_Toc500234062)

[Set ud fra systemudviklerne 10](#_Toc500234063)

[HovedAFSNIT: Software DEsign 11](#_Toc500234064)

[Historik 11](#_Toc500234065)

[Vision 12](#_Toc500234066)

[Krav 12](#_Toc500234067)

[Use case diagram 13](#_Toc500234068)

[Use cases 14](#_Toc500234069)

[Brief 15](#_Toc500234070)

[Fully dressed 16](#_Toc500234071)

[FURPS 25](#_Toc500234072)

[Funktionelt 25](#_Toc500234073)

[Brugervenlighed 25](#_Toc500234074)

[Pålidelighed 25](#_Toc500234075)

[Ydeevne 25](#_Toc500234076)

[Support 25](#_Toc500234077)

[Hovedafsnit: Software Konstruktion 26](#_Toc500234078)

[Fremgangsmåde 26](#_Toc500234079)

[Sammenligning med FURPS + argumentation 26](#_Toc500234080)

[Redegørelse af udvalgte metoder – hvor og hvorfor bruges de? + metodegenbrug (redundans, optimering) 26](#_Toc500234081)

[Hurtig forklaring om login-feature (FURPS pålidelighed) 26](#_Toc500234082)

[Switches – brugervenlighed 26](#_Toc500234083)

[Redigering – hvorfor har vi implementeret redigering i så mange aspekter af programmet? 26](#_Toc500234084)

[Gennemgang af klasser 26](#_Toc500234085)

[Delkonklusion: produkt 26](#_Toc500234086)

[Delkonklusion: arbejdsprocess 26](#_Toc500234087)

[Anden information 27](#_Toc500234088)

[Arbejdsprocessen 27](#_Toc500234089)

[Bilag 27](#_Toc500234090)

[Bilag 1: Faseplan 27](#_Toc500234091)

[Bilag 2: kanban screendump 27](#_Toc500234092)

[Glossary 28](#_Toc500234093)

[Domain 28](#_Toc500234094)

[Litteraturliste 29](#_Toc500234095)

# Indledning

*Af Casper Frost Andersen*

Valhalla IT-solutions er blevet kontaktet af svømmeklubben Delfinen, der er beliggende i en mindre by i Midtjylland.   
  
Grundet en rokering i det nylige kommunalvalg, er der blevet budgetteret en masse midler til de lokale idrætsforeninger, deriblandt Delfinen. Der er desuden også blevet gjort en masse gratis reklame for bl.a. Delfinen, for at få gang i forretningen, hvilket naturligvis har medført en stigning i antallet af henvendelser.  
  
Derfor har bestyrelsen valgt at kassere deres manuelle registreringssystem, og i stedet investere i et administrativt IT-system, der skal kunne gøre de ansattes arbejde lettere og mere overskueligt.

Da Delfinen stadig er en lille klub, som ikke har haft nogen større økonomisk tilslutning fra lokale sponsorer eller fra kommunens side, har det på nuværende tidspunkt kun været muligt at anskaffe sig en enkelt computer, uden opkobling til internettet.

Derfor ønsker bestyrelsen nu et simpelt system som kan anvendes på computeren, og som desuden lagrer filerne lokalt, grundet den manglende opkobling.

# Problemformulering

*Af Casper Frost Andersen*

Denne rapport har til formål at besvare følgende problemformulering:

*Hvordan laver vi mest effektivt et administrativt system, som opfylder de krav, som klubben Delfinen søger?*

Dette vil ske ud fra en redegørelse af følgende delpunkter:

* En analyse af svømmeklubben Delfinen som virksomhed (ITO).
* Et design-afsnit, hvori vi både skriftligt og visuelt formulerer hele og dele af vores bud på en løsning (SWD).
* Et afsnit hvor der redegøres for det system vi har lavet, samt argumentation for de løsninger vi har valgt.

Derudover forholder vi os til følgende spørgsmål:

* Kan der introduceres nogle features som tilgodeser kundens behov, og som stadig er i overensstemmelse med det oprindelige udgangspunkt?

# Afgrænsning

Suggestions

1. Redigere i medlemsoplysninger.
2. Tilføje flere discipliner og tider til medlem.
3. Oprette ansatte.
4. Automatisk restance opdatering på medlem (ud fra dato).

# Interessent analyse

*Af Rasmus Sadurski*

## Brainstorm

Fra svømmeklubben- og systemudviklernes side

Er der nogen som kommer i berøring med klubben?

Hvem er ejerne af Svømmeklubben Delfinen? Hvem har en indflydelse i svømmeklubben?

* Ledelsen
* Formanden
* Kassereren
* Træneren
* Medlemmer

Hvordan har systemudviklerne en indflydelse i svømmeklubben?

* Systemudviklerne skal lave et styresystem, der opfylder de specifikke krav ledelsen har til systemet.
* Udviklerne skal være færdige inden for en bestemt deadline.

Hvilken rolle har formanden i klubben?

* Formanden tager imod nye medlemmer, og registrerer dem med deres stamoplysninger, fx navn, alder osv.

Hvem betaler til klubben?

* Det gør medlemmerne. Kontingenterne lyder som følger:
  + Under 18 år (1000 kr.) årligt.
  + Over 18 år (1600 kr.) årligt.
  + Aktive medlemmer over 60 år får 25% rabat
  + Passivt medlemskab (500 kr.) årligt.

Hvad skal systemudviklerne være opmærksomme på mht. priserne?

* Udviklerne skal være opmærksomme på, om medlemmerne er over eller under 18 år.
* Udviklerne er opmærksomme på om medlemmerne er over 60 år.

Hvem kan blive generet af klubbens tilstedeværelse?

* Konkurrenter til klubben, altså andre svømmeklubber i nærområdet.
* Konkurrencesvømmer der konkurrerer imod Delfinerne.

Er der mulighed for, at skabe en karriere i svømmeklubben?

* Man kan gå til konkurrencesvømning, med en tilknyttet træner dertil.
* Konkurrencesvømmerne er inddelt i to hold: ungdomshold (under 18 år) og seniorhold (18 år og over).

## Sortering

Er der nogen som kan havde en indflydelse og/eller havde en påvirkning?

*Man kan sige, at jo mere indflydelse og påvirkning der er i klubben desto mere interesse er der i klubben.*

## Svømmeklubbens matrix

|  |  |  |
| --- | --- | --- |
|  | *Ikke indflydelse i klubben* | *Indflydelse i klubben* |
| *Interesse i klubben* | Andre svømmeklubber  Andre konkurrencesvømmere | Aktive medlemmer  Konkurrencesvømmere  Ledelsen  Kassereren  Træneren  Sponsorer |
| *Ikke interesse i klubben* | Er der en kantine?  Pedellen i svømmeklubben | Passive medlemskaber  Mindre klub som er i vækst  Rengøring af klubben |

## Systemudviklernes matrix

|  |  |  |
| --- | --- | --- |
|  | *Ikke indflydelse i klubben* | *Indflydelse i klubben* |
| *Interesse i klubben* | Andre systemudviklere | Os som systemudviklere  Ledelsen (et hurtigt system) |
| *Ikke interesse i klubben* | Det gamle styresystem | Skift af gamle styresystem |

## Interessenterne

Hvis man skulle udvælge de vigtigste interessenter der måtte være.

Hvad ønsker ledelsen at se fremadrettet?

* En øget vækst af medlemmer.
* Gode resultater fra konkurrencesvømmerne.
* En øget vækst i økonomien.
* At kunne udkonkurrere de andre svømmeklubber i nærområdet.

Hvad ønsker ledelsen af det nye styresystem?

* Det bliver nemmere at registrere nye medlemmer.
* Systemudviklerne overholder deadlinen.
* Systemet skal som udgangspunkt ikke fornys inden for de første fem år.

Vil der kunne opstå ulemper som ledelsen ikke ønsker at se fremadrettet?

* En faldene vækst af medlemmer.
* Dårlige resultater fra konkurrencesvømmerne.
* Nye svømmeklubber åbner.
* Faldende omsætning.

Hvad ønsker ledelsen ikke at se mht. systemet?

* At systemet ikke lever op til ledelsens krav.
* Udviklerne overholder ikke deadlinen.
* Systemet bliver for hurtigt forældet.

Hvad ønsker udviklerne af ledelsen?

* De har en klar idé om hvad systemet skal kunne.
* En fast deadline.

Hvis man skulle vurdere vigtigheden af medlemmerne i svømmeklubben (1-5)?

* Vigtigheden af medlemmer er en 5’er, for uden medlemmer, ingen svømmeklub.

## Integrer

Analyse af svømmeklubben / systemudviklerne.

Svømmeklubben skal identificere interessekonflikter, af problemer der kan opstå for svømmeklubben.   
En god idé for svømmeklubben er, hvis de laver en *risikoanalyse*, mulige *målsætninger i svømmeklubben*, og så en *kommunikationsplan*.

En risikoanalyse er en god idé, fordi man hurtig kan lave en brainstorm om, hvilke risici der kan opstå for svømmeklubben.

* Vil der opstå strike?
* Skader på bygningen og deres faciliteter!
* Samfundsøkonomi

Hvad der kunne opstå for udviklerne.

* Sygdom.
* Overholder ikke datelinen.

Hvad vil svømmeklubbens målsætninger være? Hvilken vision har svømmeklubben egentlig?

* Vil svømmeklubben prøve og etablere sig blandt, Danmarks bedste svømmeklubber?
* Har svømmeklubben en drøm om at skabe Danmarks næste store svømme ikon?

Udviklernes målsætning.

* Holde morgenmøde hver dag.
* Havde en liste over hvad er færdig.
* Være klar over, hvad der skal laves.

En kommunikationsplan tydeliggør interessenternes opmærksomhed. Folk (medlemmer og nysgerrige udefrakommende) vil gerne havde information, og ønsker nogen krav om svømmeklubben, vil de overhovedet havde nogen indflydelse i svømmeklubbens historie? ”Medlemmerne” i denne situation er interessenterne, det er dem svømmeklubben skal satse på. Hvordan skal svømmeklubben sende deres budskab ud?

Udviklernes kommunikationsplan er

* Møder op til morgenmøde hver dag.
* Færdig gøre dagens protjek.
* Være sikker på de overholder datelinen.
* Teste færdig kode.
* Brainstorm med resten.

## Interessent matrix

af Martin Løseth Jensen & Rasmus Sadurski

Positiv indstillet

Authority

Formanden

Sponsor

Kunder (klubbens ledelse)

Kasserer

Trænere

Medlemmer

Konkurrenter

Vigtig

Interesse organisationer

Uvigtig

Negativ indstillet

# SWOT-analyse

Af Rasmus Sadurski & Martin Løseth Jensen

SWOT-analysen skal hjælpe os med, at skabe struktur og overblik over svømmeklubbens styrker og konkurrencesituation. SWOT-analysen giver et virtuelt billede af, hvad svømmeklubbens aktuelle situation er.

## Set ud fra svømmeklubben

|  |  |
| --- | --- |
| Interne forhold | |
| Stærke sider (+) | Svage sider (-) |
| * Ny svømmeklub (nyhedsværdien) * Flere aktiviteter og medlemskaber1 * Central placering * Deltagelse ved sociale aktiviteter * Betaling (at pris og faciliteter hænger sammen) * Konkurrencesvømmerne har deres egen træner * Reklame | * Ny svømmeklub (ukendt, mangel på reklame) * Ikke så mange medlemmer * Dårlig placering * For lav deltagelse ved sociale aktiviteter * Betaling (at pris og faciliteter ikke hænger sammen) * Mangel på konkurrencesvømmere * Mangel på reklame/dårlig reklame2 |
| Eksterne forhold | |
| Muligheder (+) | Trusler (-) |
| * Bedre faciliteter * Udvidelse af anlæg * Konkurrencestævner * Samarbejdspartnere * Sponsorer | * Konkurrence fra andre svømmehaller * Manglende mulighed for udvidelse af anlæg * Manglende deltagelse ved konkurrencestævner/andre begivenheder * Svømmeklubbers konkurrenceevne * Mangel på samarbejdspartnere * Mangel på sponsorer |

1. Da det er en mindre svømmeklub under vækst, ønsker den selvfølgelig at få så mange nye medlemmer som muligt. Det kan svømmeklubben blandet andet gøre via reklame.
2. Det vides ikke om svømmeklubben har reklameret for sig selv, derfor vil det vær en god idé at sende en form for nyhed ud til den lokale befolkning (for at tiltrække nye medlemmer). Svømmeklubben kan evt. reklamere med konkurrencesvømning.

## Set ud fra systemudviklerne

|  |  |
| --- | --- |
| Interne forhold | |
| Stærke sider (+) | Svage sider (-) |
| * For at få erfaring1 * Gå på mod * Gode til at samarbejde * Brainstorm | * Mangle på erfaring2 * Håndtering af projektet * Finde den rigtige arbejdsmetode * Mangle på information * Manglende dynamik i team |
| Eksterne forhold | |
| Muligheder (+) | Trusler (-) |
| * Flere større projekter * God omtale * Flere/større kunder | * Andre systemudviklere tager jobbet * Dårlige omtale * Mangle på kunder * Tid kontra betaling |

1. Mere erfaring vi for i dette projekt, jo bedre bliver vi også fremadrettet. Det vil blive nemmere at få et overblik, over nye projekter. Ved flere projekter giver det også en bedre forståelse på en god arbejdsmetode.
2. Da vi ikke har så meget erfaring, hverken i kode heller designe, giver det os, nogen få problemer hvordan man skal se projektet. Hvordan skal vi håndtere projekt, hvilken arbejdsmetode bruger vi.

# HovedAFSNIT: Software DEsign

## Historik

|  |  |  |  |
| --- | --- | --- | --- |
| Dato | Version | Forfatter | Beskrivelse |
| 20-11-2017 | 00.01.01 | Hele gruppen | Første sammensætning af OOAD. |
| 21-11-2017 | 00.02.01 | Hele gruppen | Implementering af OOAD i rapport. |
| 24-11-2017 | 00.02.02 | Hele gruppen | Tilføjelse af UC007 + SSD, tilføjelse til SWOT. |
| 24-11-2017 | 00.02.03 | Hele gruppen | Tilføjelse af arbejdsprocesdel. |
| 27-11-2017 | 00.02.04 | Hele gruppen | Rettelser i nuværende use cases, SSD’er Tilføjet: UC009 SD Use case diagram |
| 28-11-2017 | 00.02.05 | Hele gruppen | Print af restance list Print betalingsstatus list Register konkurrencesvømmer til stævner  Gemmer til stævner fil  Printer konkurrencesvømmer i udvalgte disciplin  US001 US002 US008 |
| 29-11-2017 | 00.02.06 | Hele gruppen | Testning af SWC  Update af SWD  Rettelser/update af Faseplan |
| 30-11-2017 | 00.02.07 | Hele gruppen | Redigér kontingent  Format på console og når det gemmes i fil Opfylder krav for aktiv og passiv  UC004 top 5 |
| 1-12-2017 | 00.02.08 | Hele gruppen | Opdatering af UCD, tilføjelse af 3 brief UC’s Print stævneliste Kontintentpris Print junior og senior hold |
| 4-12-2017 | 00.03.01 | Hele gruppen | Ret rapport  SD Formatering Træner  Snippets af kode til rapport |
| 4-12-2017 | 00.03.02 | Hele gruppen | Tilføjet udkast til indledning og problemformulering. |
| 5-12-2017 | 00.03.03 | Hele gruppen | Tilføjet hovedafsnit: konstruktion |

## Vision

Af Casper Frost Andersen

Vi ønsker at lave et administrativt program til svømmeklubben Delfinen. Programmet skal håndtere indmelding af nye medlemmer samt deres specifikationer, herunder deres personlige data, som alder, medlemstype, og aktivitetsniveau (motionist/konkurrence).

Herudover skal medlemmernes kontingenter registreres ved indmeldelse. Kontingentet er reguleret af flere variabler, herunder alder, aktivitetsform og type af medlemskab. En kasserer skal kunne se en oversigt over medlemmer i restance.

En træner, der er tilknyttet svømmeklubben, skal kunne se en oversigt over deltagerne på sine hold, samt deres discipliner og bedste tider, indenfor disse.  
Ydermere skal træneren kunne få printet en top 5 oversigt over de bedste konkurrencesvømmere inden for de forskellige discipliner, i forbindelse med udtagelse til stævner.

## Krav

Af Casper Frost Andersen

* Systemet skal kunne lagre data af forskellige typer fra forskellige brugere.

Overordnet er der tre typer af data, der skal gemmes:

* + Medlemmernes informationer
  + Kontingenter og restancer
  + Medlemmernes svømmetider
  + Data relateret til stævner, hvor stævnernes navne, og svømmernes tider indgår.
* Formanden vil være den eneste, der kan oprette nye medlemmer i svømmeklubben. De nye medlemmer skal opgive deres navn, type medlemskab (aktiv/passiv), alder og aktivitetsform (motionist/konkurrencesvømmer).
* Kassereren håndterer alle kontingenter og restancer. Vedkomne skal være i stand til at udskrive en oversigt over medlemmer i restance. De forskellige typer abonnementer er:
  + Aktiv:
    - Under 18: 1000 kr.
    - Over 18: 1600 kr.
    - Senior over 60 får 25% rabat.
  + Passivt: 500 kr.

Alle gebyrer indbetales årligt.

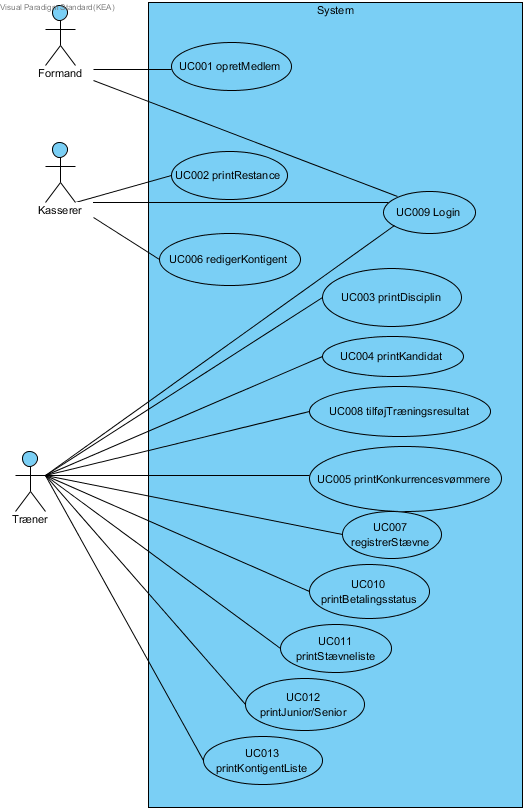
* Træneren har to hold: en for medlemmer under 18 år (ungdomshold) og en for over 18 (seniorhold).

­Hvert medlem har deres disciplin(er) tildelt i datafilen.   
Hver svømmer får deres bedste tid registreret, og den dato den blev opnået.   
Hvis nogen af medlemmerne konkurrerer til et stævne, gemmes navnet på stævnet sammen med svømmernes placering og tid.  
Træneren skal være i stand til at udskrive en top 5 af svømmerne i alle discipliner, med henblik på udtagelse til stævner.

Disse er:

* + Butterfly
  + Crawl
  + Rygcrawl
  + Brystsvømning
  + Hundesvømning

## Use case diagram



## Use cases

Af Martin Løseth Jensen & Christian Strunge

### Brief

#### Use Case 004

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 004 |
| Use Case Navn | Print Udtagelseskandidater |
| Main Succes Scenario | Træner skal kunne se en liste over den nuværende top 5 (de 5 bedste tider) inden for hver enkel disciplin. |

#### Use case 005

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 005 |
| Use Case Navn | Print konkurrencesvømmere |
| Main Succes Scenario | Træner skal kunne printe en liste over alle konkurrencesvømmere. |

#### Use case 006

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 006 |
| Use Case Navn | Rediger betalingsstatus |
| Main Succes Scenario | Kassér skal kunne ændre medlemmers kontingent. |

#### USE CASE 7

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 007 |
| Use Case Navn | Registrer stævne |
| Main Succes Scenario | Træneren skal kunne registrere konkurrencesvømmere som har deltaget i stævner. |

#### Use case 9

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 009 |
| Use Case Navn | Vælg ansat |
| Main Succes Scenario | Brugeren af systemet skal have mulighed for at logge ind som en bruger |

#### USE CASE 10

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 010 |
| Use Case Navn | Print betalingsstatus |
| Main Succes Scenario | Kasserer skal kunne printe en liste over alle medlemmer samt deres betalingsstatus |

#### USE CASE 11

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 011 |
| Use Case Navn | Print stævneliste |
| Main Succes Scenario | Træner skal kunne se en liste over samtlige konkurrencesvømmere, der har deltaget i stævne med tid og placering |

#### USE CASE 12

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 012 |
| Use Case Navn | Print junior/senior hold |
| Main Succes Scenario | Træner skal kunne printe 2 hold ud, hhv. junior- og seniorhold, fra konkurrencesvømmerlisten. |

#### Use case 13

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | Use Case 013 |
| Use Case Navn | Print et medlems kontingentpris |
| Main Succes Scenario | Kassereren skal kunne se en liste over medlemmer, vælge et medlem, og se dennes kontingentbetaling. |

### Fully dressed

#### Use case 001

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case ID | UC001 |
| Use Case Navn | Opret medlem |
| Scope | Svømmeklub Delfinen administrativt system |
| Beskrivelse | Formand skal kunne oprette nye medlemmer, og tilknytte staminformation og aktivitetsform. |
| Aktør(er) | Formand |
| Stakeholders | Formand for svømmeklubben Delfinen |
| Preconditions | Nyt medlem vil oprettes i klubben |
| Succes garanti | Medlemmets informationer bliver gemt i systemet |
| Main flow | 1. Formanden vælger *opret nyt medlem* fra en menu. 2. Systemet prompter for aktivitetsform 3. Formand indtaster aktivitetsform 4. Systemet prompter for fornavn 5. Formand indtaster fornavn 6. System prompter for efternavn 7. Formand indtaster efternavn 8. System prompter for alder 9. Formand indtaster alder 10. System prompter for medlemstype 11. Formand indtaster medlemstype 12. Systemet prompter for disciplin 13. Formand indtaster disciplin 14. Systemet gemmer medlemmet i fil. |
| Extensions | Gældende for trin 3, 5, 7, 9, 11, 13:   1. Formand indtaster ugyldigt input 2. Hvis input er gyldigt, fortsæt da til næste trin. Ellers returner til samme trin.   3a. Formand vælger motionist   1. Følger trin 4 til 11. 2. Skipper trin 12 & 13 3. Systemet gemmer medlem i fil. |

##### UC001 SSD



##### Sequence diagram UC001

Et billede, der indeholder skærmbillede

Beskrivelse, der er oprettet med meget høj sikkerhed

#### use case 002

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case Navn | Print svømmediscipliner |
| Use Case ID | UC002 |
| Scope | Svømmeklub Delfinen administrativt system |
| Beskrivelse | Træner skal kunne se en eller flere lister over svømmere, der er tilknyttet de forskellige svømmediscipliner. |
| Aktør(er) | Træner |
| Stakeholders | Træner for svømmeklubben Delfinen |
| Preconditions | Klubben har medlemmer der er konkurrencesvømmere.  Medlem skal være tilknyttet en disciplin. |
| Succes garanti | En liste over svømmere og de tilknyttede discipliner bliver printet. |
| Main flow | 1. Træner vælger *print discipliner* fra træner menu. 2. System prompter for disciplin. 3. Træner vælger disciplin. 4. System printer liste 5. System prompter om returnering 6. Træner vælger at returnere 7. System returnerer til ”træner menu” |
| Extensions | 3a. Træner indtaster ugyldigt input.   1. System prompter for gyldigt navn for disciplin. 2. Hvis navn er gyldigt, fortsæt til punkt 4. Ellers returner til 3a   5a. Træner vælger at printe en ny liste ud.   1. Returner til punkt 2.  Ellers fortsæt til punkt 6. |

##### UC002 SSD



#### Use Case 003

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case Navn | Print restancer |
| Use Case ID | UC003 |
| Scope | Svømmeklub Delfinen administrativt system |
| Beskrivelse | Kassereren skal kunne se en liste over de medlemmer der er i restance. |
| Aktør(er) | Kasserer |
| Stakeholders | Kasserer for svømmeklubben Delfin |
| Preconditions | Klubben har mindst et medlem |
| Succes garanti | En liste over medlemmer der er i restance bliver printet ud på konsollen. |
| Main flow | 1. Kasserer vælger *print restanceliste* fra menu. 2. System printer liste. 3. System returnerer til menu. |
| Extensions | 1a. Ingen medlemmer er i restance.   1. System printer besked og fortsætter til punkt 3. |

##### UC003 SSD



#### Use Case 8

|  |  |
| --- | --- |
| Use Case Section | Kommentar |
| Use Case Navn | Opret ny bedste tid |
| Use Case ID | UC008 |
| Scope | Svømmeklub Delfinen administrativt system |
| Beskrivelse | Træner skal kunne tilføje træningsresultat og dato til konkurrencesvømmere, og med mulighed for gentagelse |
| Aktør(er) | Træner |
| Stakeholders | Træneren i svømmeklubben Delfinen |
| Preconditions | Klubben har mindst én konkurrencesvømmer |
| Succes garanti | Et træningsresultat og dato for denne er blevet tilknyttet en eksisterende konkurrencesvømmer |
| Main flow | 1. Træner vælger *tilføj træningsresultat* fra *træner menu* 2. System printer liste over eksisterende konkurrencesvømmer(e) 3. Træner vælger svømmer fra liste 4. Systemet prompter for træningstid 5. Træner indtaster træningstid 6. Systemet prompter for dato 7. Træner indtaster dato 8. System prompter om fortsættelse 9. Træner vælger at fortsætte 10. System returnerer til *træner menu* |
| Extensions | 3a. Træner giver ugyldigt input   1. prompt for gyldigt input 2. Hvis input er gyldigt, fortsæt til punkt 4   5a. Træner giver ugyldigt input   1. prompt for gyldigt input 2. hvis input er gyldigt, fortsæt til punkt 6   7a. Træner giver ugyldigt input   1. prompt for gyldigt input 2. hvis input er gyldigt, fortsæt til punkt 8   9a. Træner vælger at indtaste nyt træningsresultat   1. returner til punkt 2 |

##### UC008 SSD



## FURPS

Af Casper Frost Andersen

### Funktionelt

* Systemet skal kunne gemme data om:
  + Medlemmer og deres medlemskabstyper og aktivitetsform.
  + Kontingenter og restancer for medlemmer.
  + Medlemmernes bedste tider i de forskellige discipliner.
  + Oversigt over konkurrencesvømmeres deltagelse i stævner (Placering, bedste tid og stævnets navn).
* Formanden i klubben skal være i stand til at oprette nye medlemmer i svømmeklubben.
* Kassereren skal kunne se et udprint over medlemmernes restancer i programmet.
* Træneren skal være i stand til at se et overblik over de fem bedste svømmere i alle discipliner, med henblik på udtagelse til stævner.
* Systemet skal kunne gemme data på lokale filer.

### Brugervenlighed

* Forskellige brugere kommer til at have forskellige funktioner i systemet (formanden kan for eksempel oprette nye medlemmer, men vil ikke have adgang til de økonomiske funktioner).
* Systemet skal være brugervenligt.
* Bruger skal kunne hente tilgængeligt indhold fra lagrede data.

### Pålidelighed

* Systemet vil løbende indsamle nye data, som brugerne indtaster. Disse bør bakkes op på eksterne lagerenheder en gang pr. uge, i tilfælde af et system-breakdown eller andre hændelser, der kan resultere i tab af data.
* Medlemmerne data skal opbevares på en password beskyttet computer.

### Ydeevne

* Systemet skal skrives uden redundans og unødvendige beregninger, for at sikre optimal program-hastighed.
* Programmet kræver ikke meget hukommelse, så programmet bør fungere optimalt på alle nyere computere.

### Support

* Programmet skal være letlæseligt og skal kunne tilpasses, hvis kunden ønsker nye funktioner tilføjet.
* Programmet skal være veldokumenteret, så det er let at vedligeholde.

# Hovedafsnit: Software Konstruktion

## Indledning

Dette afsnit har til formål at redegøre for selve programmeringsprocessen. I afsnittet vil vi komme ind på vores fremgangsmåde, en overordnet gennemgang af koden, eksempler fra source koden og argumentation for, hvorfor vi har valgt netop de løsninger, og til sidst en redegørelse af de ekstra features vi har implementeret.

## Fremgangsmåde

Vi har fra starten aftalt at holde møder dagligt, hvor vi forventningsafstemmer i forhold til hvor langt vi er nået i opgaven – har vi nået det vi skulle, har vi fået nogle gode idéer, og hvad vi skal lave fremadrettet. Derudover har vi også holdt mandags- og fredagsmøder, hvor vi hhv. redegør for ugens delmål, og efterfølgende opsummerer og reflekterer over ugens forløb.

Vi har taget udgangspunkt i de Use cases vi har lavet i design-delen, og iterativt tilføjet dem til programmet. Efter hver implementering har vi testet og diskuteret, om koden passer til den allerede eksisterende kode. Efterfølgende har vi reflekteret over suggestions, afgrænsning, og om vi kan fjerne redundans.

## Use cases til konstruktion\*\* (temp)

* UC001 – opret
* UC002 - print
* UC006 - rediger

## Hurtig forklaring om login-feature (FURPS pålidelighed)

Vi har jf. vores pålidelighedsafsnit i FURPS tilføjet en login-funktion til programmet. Den er lavet ved en simpel switch i et while-loop. Funktionen giver muligheden for at ”logge ud” af programmet, som man kender det på en computer, og derved forhindrer ”ubudne gæster”

## Switches – brugervenlighed

Vi har taget udgangspunkt i switches igennem hele vores kode, for at øge brugervenligheden og mindske fejlmarginen ved indtastning. Det simplificerer brugerfladen, og gør det derudover også muligt at tilføje returneringsfunktioner i de forskellige undermenuer. Derved gøres det lettere at finde rundt i menuerne, og forhindrer at programmet skal køres fra start til slut hver gang.

## Sammenligning med FURPS + argumentation

## Redegørelse af udvalgte metoder – hvor og hvorfor bruges de? + metodegenbrug (redundans, optimering)

* subMenu()
* testConsoleInput()
* junior()
* redigerKontingent()

## Redigering – hvorfor har vi implementeret redigering i så mange aspekter af programmet?

Det har vi fordi at det er at det har vi

# Delkonklusion: produkt

# Delkonklusion: arbejdsprocess

### Arbejdsprocessen

Vi har i løbet af dette forløb benyttet os af en iterativ arbejdsproces. Dette betyder at vi hver gang en del af opgaven er blevet udviklet eller færdiggjort, reflekterer gruppen over, om det passer ind i den øvrige opgave, og i det overordnede billede.

Vi har arbejdet ud fra nogle grundprincipper, der lyder som følgende:

* Forventningsafstemning – sæt realistiske mål
* Kommunikation – løbende møder hvor vi gør status over, hvad vi har lavet indtil videre, hvor vi er i opgaven, hvor vi vil hen, og om vi følger arbejdsplanen.
* Oplæg – løbende gennemgang af kode, diagrammer og layout, så alle har samme forståelse for forløbet.

Med henvisning til bilag 1, fremgår det, at vi løbende har tilføjet arbejdsopgaver for at imødekomme vores mål. Hvis vi har været forud for faseplanen, har vi justeret derefter, og vice versa.

# Konklusion

# Bilag

## Bilag 1: Faseplan

## Bilag 2: kanban screendump

# Glossary

Martin Løseth Jensen

### Domain

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Aliases | Labels | Objects |
| Formand |  | Aktør | Administrativ ansat |
| Træner |  | Aktør | Ansat der træner konkurrencesvømmere |
| Kasserer |  | Aktør | Ansat der holder styr på kontingentbetaling |
| Pay |  | Association | Betaling |
| Uses |  | Association | Bruger |
| Attends |  | Association | Deltager |
| Record |  | Association | Gemmes |
| Contain |  | Association | Indeholder |
| Contained in |  | Association | Indeholdes i |
| Initialize |  | Association | Opret |
| Results |  | Association | Resultat |
| Aktivitetsform |  | Objekt | Den form for aktivitet der udøves |
| Disciplin |  | Objekt | Den specifikke aktivitet der konkurreres i |
| Konkurrence |  | Objekt | En forsamling af konkurrencesvømmere der konkurrerer i forskellige discipliner. |
| Medlem |  | Objekt | En person som er indmeldt i klubben |
| Hold |  | Objekt | Betegnelse for flere medlemmer der svømmer sammen |
| Kontingent |  | Objekt | Medlem skal betale dette for at oprettes og forblive i klubben |
| Restance |  | Objekt | Når kontingent ikke er betalt til tiden |
| Prompt |  | System | Kort besked fra systemet der beder om indtastning |

# Litteraturliste

Craig Larman: Applying UML and Patterns: An introduction to Object-Oriented Analysis and Design and Iterative Development – Third edition (2004)

Building Java Programs \*\*\*\*\*\*

ITO-bogen\*\*\*\*\*\*\*\*\*

[www.stackoverflow.com](http://www.stackoverflow.com)

søgemaskinen Google.