

Försök att kompilera / använda källkoden enligt instruktionerna.

- Kan du få igång den?
Ja.
- Är det något problematiskt?
Njae, vissa funktioner fungerar inte..
- Finns det steg som saknas?
Ja, eller njae. Ta bort och redigera båt finns i listan men det fungerar inte..
- Visar diagrammen och implementationen samma sak?
Diagrammet visar detsamma som implementationen fast det saknas en relation mellan ClubData och viewmessage i klassdiagrammet, skall vara en dependency mellan dem.
- Finns det några saknade relationer?
ja, mellan ClubData och ViewMessage.

Är akritekturen ok?

- Finns det en modell / vy-separation
Nej. Finns modeller och vyer men member modellen skapar strängar som den returnera och clubData skapar en vy så de går emot allt med MVC.
- Är modellen kopplad till användargränssnittet?
Ja det är den.
- Är modellen specialiserad för en viss typ av UI? T ex för att returnera formaterade stängar som ska skrivas ut.
Alla modeller returnerar strängar och ena modellen skapar en vy.
- Finns det domänregler i gränssnittet?
Finns inga regler i gränssnittet.
- Är kravet på ett unikt medlemsnummer korrekt gjort?
 - 1.Nej, vi testade att ändra medlemsnummer på 2 personer till samma och det fungerade. Man ska väl inte kunna ändra det manuellt? I sådana fall borde det finnas checkar som kollar att det UNIKA medlemsnummret inte redan finns?
 2. Lägga till båt har hand om medlemsnummer. skriver du in nummer 2 som ligger på två personer just nu så får båda samma båt.
 3. Båten är kopplad till medlemsnummer. Byter man medlemsnummer från 35468 till 1 på medlemmen så försvinner båten
 - 4.Skapar man en ny medlem och ändrar medlemsnummret till 35468 som en annan medlem har, så får den här nya medlemmen automatiskt de båtar som orginalmedlemmen 35468 äger.
 5. Tar man bort medlemmen med nummer 35468 och sen skapar en ny användare nr 35468 får den här personen också en båt. Dvs båtar försvinner inte om medlemmen försvinner.
Men skapandet är okej, bara det att man kan ändra det och att man inte kollar om det redan existerar.

Hur är kvaliteten på genomförandet / källkoden?

- Kod-standarden

Koden i sig är väldigt komplicerad så kan inte se om det finns upprepad kod (DRY), däremot är vissa delar bra skrivet och vissa delar hade man nog kunnat faktorerat om.

- Namngivning

Bra namngivning.

- död kod osv osv

Nej finns ingen död kod vad vi har hittat.

Hur är kvaliteten på designen?

- Är den objektorienterad?

Ja det är objektorienterat

- Är objekt kopplade via associationer och inte med nycklar/ID'n

Både ja och nej. Båt och medlem är kopplade med IDn och inte association då man behöver veta en medlemsnummer för att kunna skapa en båt eller registrera eller ta bort.

- Är GRASP använt på rätt sätt?

Nej det är inte rätt använt, för det första går de emot MVC strukturen. Det som kan vara använt på rätt sätt i GRASP är controller delen, för att de hanterar "the overall system" i den.

- 1.** Represents the overall "system," a "root object," a device that the software is running within, or a major subsystem (these are all variations of a *facade controller*).

[applying uml patterns 3rd ed. PDF sid.922]

- Klasser har hög sammanhållning och är inte för stora eller har för mycket ansvar
Klasserna har låg sammanhållning för det mesta av funktionerna ligger i clubData (1185 rader kod i den klassen).

**High Cohesion
(evaluative)**

How to keep objects focused, understandable, and manageable, and as a side-effect, support Low Coupling?

Assign responsibilities so that cohesion remains high. Use this to evaluate alternatives.

[applying uml patterns 3rd ed. PDF sid.922]

- **Klasser har låg koppling och är inte alltför anslutna till andra enheter**
Alla klasser förutom clubData har låg koppling till varandra, clubData har koppling till alla förutom 2 klasser.

**Low Coupling
(evaluative)**

How to reduce the impact of change?

Assign responsibilities so that (unnecessary) coupling remains low. Use this principle to evaluate alternatives.

[applying uml patterns 3rd ed. PDF sid.922]

- **Unvik statiska variabler eller operatorer samt globala variabler**
Ja, inga globala eller statiska variabler.
- **Undvik dolda beroenden**
Båt och medlem har ett dolt beroende då du måste veta medlemsnumret, och man kan ändra numret vilket gör att detta beroende är rätt så allvarligt
- **Information bör vara inkapslad**
Ja, informationen är privat och inkapslad.
- **Är inspirerad från domänmodellen**
Ja det är den.
- **Finns det primitiva datatyper som egentligen borde vara klasser?**
Ja, ClubData innehåller metoder som inte borde finnas där. till exempel GetMemberData som egentligen skulle varit en Vy.

Skulle diagrammen hjälpa dig som utvecklare? Varför / varför inte

Klassdiagrammet är bra och väldigt förståeligt. Det visar sina associationer klart och tydligt och även vissa viktiga funktioner och attribut.

Vilka är de starka punkterna i designen / implementationen? Vad är riktigt bra och varför?

En bra sak är att den inte kraschar, den fångar felen och kastar undantag vilket gjorde oss glada. Grunden är hyffsad men finns en hel del som bör ändras.

Vilka är svagheter i design / implementationen? Vad bör ändras och varför?

Man kan ändra medlemsnummer, det ska man inte kunna göra och 2 användare kan ha samma medlemsnummer.

Båtar registreras på medlemsnummer så ändrar man medlemsnummer försvinner båtarna för den personen, men skapar du en ny användare och ge den samma medlemsnummer som den förra hade, så får den nya den andre personens båtar.

Man kan inte ta bort båtar heller och inte ändra information om dem. Dessutom finns det vy-hantering i modellerna och det skall inte ligga där utan i vyerna, i alla fall enligt MVC.

Tror du att design / implementationen har klarat de kriterier som finns för att passera betyg 2?

Nej vi tror inte den klarar kriterierna för betyg 2. För det första så fungerar inte viss funktionalitet och vissa saker som borde ligga i en vy ligger i modellerna och det är emot MVC-strukturen.

Testa den körbara versionen av programmet på ett realistiskt sätt.

- Notera eventuella problem/buggar.
- Radera båt

```
Namn: martin
Efternamn: arvidsson
Medlemsnummer 35468
Antal båtar: 1

Båttyp: Motorbåt
Längd: 5
Tryck på valfri tangent för att fortsätta

-----
Båtklubb
-----

1. Registrera medlem
2. Medlems lista
3. Information om enskild medlem
4. Redigera medlem
5. Ta bort medlem
6. Registrera båt
7. Ta bort båt
8. Redigera båt
0. Avsluta och sparar alla ändringar som gjorts

-----

Fyll i menyval 0-8:
?
Vilken båt vill du radera?
Sök båt via medlemsnummer
<0 för att avbryta>
35468
Kunde inte hitta den båten, försök igen
Tryck på en tangent för att fortsätta...
```

Vilken båt vill du radera?

Vi vill radera båt på medlemsnummer 35468.

Vi matar in nummret

Får tillbaka:

Kunde inte hitta den båten försök igen

* redigera båt - samma sak här

```
Namn: martin
Efternamn: arvidsson
Medlemsnummer 35468
Antal båtar: 1

Båttyp: Motorbåt
Längd: 5
Tryck på valfri tangent för att fortsätta
2
-----
Båtklubb
-----

1. Registrera medlem
2. Medlems lista
3. Information om enskild medlem
4. Redigera medlem
5. Ta bort medlem
6. Registrera båt
7. Ta bort båt
8. Redigera båt
9. Avsluta och sparar alla ändringar som gjorts
-----

Fyll i menyval 0-8:
8
Vilken båt vill du redigera?
Sök båt via medlemsnummer
<0 för att avbryta>
35468
Kunde inte hitta den båten, försök igen
Tryck på en tangent för att fortsätta...
```

Case 2

1. Vi har registrerat två nya medlemmar

Kristoffer	Medlemsnummer: 10695
2 båtar	Segelbåt och Roddbåt

Emilia	Medlemsnummer: 66155
0 Båtar	-----

Vi avslutar och sparar ändringarna sen vill vi redigera Emilia

```
Namn: Kristoffer
Efternamn: Svensson
Medlemsnummer 10695
Antal båtar: 2

Båttyp: Segelbåt
Längd: 10
Båttyp: Roddbåt
Längd: 3

Namn: Emilia
Efternamn: Drake
Medlemsnummer 66155
Antal båtar: 0
```

```

Namn: Emilia
Efternamn: Drake
Personnummer: 890419
Medlemsnummer: 66155

Vilken egenskap vill du ändra?

0. Avsluta
1. Förnamn
2. Efternamn
3. Personnummer
4. Medlemsnummer

```

Vi vill ändra Emilias medlemsnummer till hennes lyckotal, 10695. Ovetandes om att det finns en annan medlem som har detta medlemsnummer.

```

Namn: Emilia
Efternamn: Drake
Personnummer: 890419
Medlemsnummer: 66155

Vilken egenskap vill du ändra?

0. Avsluta
1. Förnamn
2. Efternamn
3. Personnummer
4. Medlemsnummer
4
Vad vill du byta 66155 till?

```

Kollar vi sen i den detaljerade listan så ser vi att två medlemmar har samma "unika" medlemsnummer. Dessutom har Emilia fått samma båtar som Kristoffer har.

```

Namn: Kristoffer
Efternamn: Svensson
Medlemsnummer: 10695
Antal båtar: 2

Båttyp: Segelbåt
Längd: 10
Båttyp: Roddbåt
Längd: 3

Namn: Emilia
Efternamn: Drake
Medlemsnummer: 10695
Antal båtar: 2

Båttyp: Segelbåt
Längd: 10
Båttyp: Roddbåt
Längd: 3

```