

Hej, här kommer lite kommentarer för workshop 2.

KLASSDIAGRAM

Det ser bra och enkelt ut, men jag ser att direkt model-vy kopplingar görs mellan member och boat vilket är mindre bra.

SEKVENSDIAGRAM

Controller-klasserna är inte korrekta då memberView går direkt till boat-klassen och member-klassen,

istället för att gå via memberlist.

I övrigt ser dem bra ut.

TEST

Programet var lätt att hitta och fungerar enkelt och bra, inte speciellt grafiskt tilltalande men mycket funktionellt. Jag saknar dock båtnamn. någon typ av bekräftelse på att man ha gjort klart en inmatning som tex "member added", när en medlem har lagts till vore trevligt.

Båttyp går ju att skriva in lite vad som helst, kanske borde man få några alternativ istället.

Att man kan lägga till en båt innan någon medlem registrerad är en bug som nog bör fixas.

KÄLLKOD

Förutom att den inte stämmer överrens med klassdiagramet och sekvensdiagramet då memberView-klassen går direkt till

Boat och member -klasserna och begär ut information så ser den bra ut.

Mycket välgjord, naturliga variabelnamn och ingen onödig kod.

Nu till frågorna:

1. As a developer would the diagrams help you and why/why not?

- Yes, but they could have been done better, with a clear notation if the class is model, view or controller.

The model-view separation has flaws.

2. What are the strong points of the design/implementation, what do you think is really good and why?

The code is great, it's clean and simple and probably takes very little space.

3. What are the weaknesses of the design/implementation, what do you think should be changed and why?

A controller class should ask the memberlist for the boat and member information, not the view class directly.

4. Do you think the design/implementation has passed the grade 3 criteria?

Yes, but I think it will pass. If you fix the model-view separation it will definately pass.

//Magnus Nilsson