Filip Rydberg
Kim Nilsson
Mathias Sundin

Workshop II **by Daniel Vedin**

Peer Review

Is the requirement of a unique member id correctly done?
Yes, it's a smart and easy solution for an unique ID.

What is the quality of the implementation/source code?
The code as easy to follow and understandable with the naming. We couldn't find any bugs except when we started tinkering in the txt files manually. We think this is because of row dependency, or something similar. The other only bug we could find was that the social security number didn't have proper regex for swedish personal number. As long as you enter twelve numbers, it passed. But that's no biggie.

As a developer would the diagrams help you and why/why not?
Yes. The sequence diagrams are easy to understand, we would skip to mention the writelines in the view though, since they are not necessary.

Overall you have done three good and helpful diagrams, in other words, it is hard to spot eventual errors.

What are the strong points of the design/implementation, what do you think is really good and why?
As said above, overall code structure was okey and easy to follow. The naming was good and understandable. The MVC had good structure except that we think you should have a separate class for handling file read/write, as said in next question

What are the weaknesses of the design/implementation, what do you think should be changed and why?
We found that you write to two different files when you want to save boats and members. This also occurs in two separate classes. You could do this in one master-file/class.

You should convert the text into objects in the code and not read the information row by row.

Do you think the design/implementation has passed the grade 2 criteria?
Yes, the requirements are fulfilled and the code compiles.