Peer review John Steczko

There was no runnable version.

The source code could not be run directly, you had to change source compliance. There was no read me file that mentioned this. The changes made in the program is never saved to a database. So when we started the program again all changes was gone.

There was no class diagram.

The UniqueID is randomized and is compared to previous members ID and if it already exists no member is created. To fix this problem either choose a unique id directly or randomize until a valid unique id is found.

There are multiple locations in the code where you have a blank line between every line of code. This makes the code much harder to read and unnecessary long. Otherwise the quality is good.

The naming is mostly good but some methods are named numerically and that is not pretty. For instance writer1 and writer2.

There was no duplication or dead code that we could find.

There no unnecessary static variables or methods. There were some problem with encapsulation. For instance when the program asked for a boat type there were 4 suggestions but whatever you wrote was accepted.

There were no class diagram to see how the different classes were connected and although the sequence diagrams displays how every action works that is not enough for a developer to understand the program.

The best part of the implementation is that you have implemented the MVC pattern well.

The overall design is very good and the console layout is very user friendly.

The biggest weakness of the implementation was that the information was never saved to a database.

The biggest weakness of the design was that there was no class diagram.

After a class diagram is created and the save method is functional we believe the design/implementation passes the grade 3 criteria.

By: Jesper Holmström, Eleonor Lagerkrants and Marcus Darvelid