Peer review

Reviewer: Max Decroy Strandlund ms224cq Reviewed: folep02

Compiling and running application

The program worked well and I found no errors or bugs while testing it. I got impressed by the id number control.

UML diagrams

The diagrams looks good and I find no missing relations. There could be more functions written out, to make it even clearer. See Larman [1, chapter 16]

Architecture

The architecture seems fine and there is a clear model/view-separation.

Implementations

the implementation seems mostly fine and follows coding standards.

Within the Boat-class, boatType is a string but should be an enum.

Some functions returns null instead of exceptions. Example: *Console.java*, line 87. If no member is found, null is returned. The if-statement should test for exceptions, and `throw new Exception()` if something is wrong.

The Controller class is a single long while-statement, with nested if-statements. This might make it hard to read. Maybe try to have one public function `Show()`, and instead of if-statements, have multiple, separate private functions?

Design

The design is good and follows the grasp-principles well. [1, p. 435]

There might be a hidden dependency to the database format in the models, buth this might also just be me not knowing Java well enough. If there is a way to remove the `@XmlElement` from the models, the model would be completely independent from the database format. But on the other hand, this might be best practice for implementing xml in Java, and then it obviously is OK to keep it as it is.

Besides that, there is a clear separation between the Database class and Controller class, but still the methods in the Database-class has XML in their names. They could be renamed to saveToDB() and loadFromDB(), that would make it easier for someone to replace the database without effecting the Controller.

Conclusion

The code is well structured and the classes is separeated and have clear responsibilities. The only weakness might be that some of the code could be refactored to make it easier to read for other coders, but that does not affect the structure or robustness of the application.

I think the application should pass.

References

1. Larman C., Applying UML and Patterns 3rd Ed, 2005, ISBN: 0131489062