

Workshop 2

Peer review for Robert Hallink's group

By Juhani Aavanen and Eddy Proca

Test the runnable version of the application in a realistic way. Note any problems/bugs.

The user can choose between English and Swedish, which is a nice feature. The console doesn't clear on page changes and the commands are all on one line which makes it difficult to get a clear view of the information. The user can enter any value as a boat type. Adding a new member is a bit confusing as to what information is required. Boats can exist without any owners. The app crashes when trying to delete a member using the ID and when using the personal number, the wrong member is deleted. Editing members doesn't work either because it edits the wrong member. Adding users with more than one name and one surname results in the name counting as a personal number. The app sometimes becomes buggy when entering too much information due to using space as the command-separator.

Try to compile/use the source code using the instructions provided. Can you get it up and running? Is anything problematic? Are there steps missing or assumptions made?

There is an abundance of instructions and information and even a troubleshooting guide. Using the source code requires that we google java knowledge.

Does the implementation and diagrams conform (do they show the same thing)? Are there any missing relations? Relations in the wrong direction?

Member is a field in Boat in the class diagram but is not required in the implementation since a boat can exist without a member. The diagram doesn't show that a boat can be assigned to a member.

Is the requirement of a unique member id correctly done?

It is hard to know which value is the unique member ID since both the ID and the personal number are possible to use when deleting or editing members. Furthermore, the edit and delete features do not work which makes it hard to tell if the requirement is met.

What is the quality of the implementation/source code?

The code seems to follow modern code standards but it's hard to assess without more knowledge of Java. The variables and methods are well named and the code is well commented. There is a lot of duplicated code in the language implementation and there are many string

dependencies between the view and the controller. There is also a lot of error presentation code that seems unnecessary.

What is the quality of the design? Is it Object Oriented?

The code is well organised and object oriented. Allowing boats to have no owners may not be inspired by the domain model.

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

The class diagram helped us get an overview of the code but some things didn't seem to match the implementation. We also found no sequence diagrams.

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

There are a lot of comments which helps when you don't know Java.

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

There are many bugs and critical features do not work. There is some duplicated code that could be cleaned up. The app crashes during normal usage.

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

Since the CRUD functionality for members is not working, the requirements for grade 3 were not met.