

Workshop 2 - Peer review on Christer Hamberg

Hi. In the following cases we found things you could perhaps improve upon in your code and design:(Keep in mind that it is only suggestions and things we think might be beneficial to improve on could be wrong)

Your UML class diagram

We think you should add more methods to your classes. This would probably make it more easier for anyone that wants an overview of your project when only looking at the diagrams. Also, adding more attributes/fields could also be beneficial. For example, on your YachClubController you have alot of methods and attributes but almost non of them are displayed in the class on the class diagram. Maybe adding all the methods and attributes in that class would give better understanding what that class entails. Last thing: The UML association notation. Wouldn't it be better if perhaps you added a 1 at the start of the CacheStorage to the Member class? Same with the boat.

The sequence diagrams

1. CreateNewMember

This line: `int result = cache.storeNewMember (name,socSecId);` it says you go through the UI, but it isn't displayed on in your diagram. Perhaps there should be a connection between the UI and cache: CacheStorage? This also go for the other methods that goes through the UI and then is implemented in the CacheStorage class. It could possibly give the diagram a better detailed overview of what is required to execute a certain requirement.

2. Display Compact List

Regarding the [all hashkeys], since the name, id etc is from the member class, wouldn't it better if you have a line association to a member class as well? Instead of only making an association from the yachtcontroller? Example: Draw a line from the cache:ChacheStorage class to a member

class, and then just add the method that are needed to get the attributes from the member class.

3. Register Boat

It follows the sequences required to make this requirement possible.

The implementation

I never used the controller in my implementation, but looking at your code, I think you have managed to get it right. There is one thing though, I can't see a test file where you test your code? Also, in your cachestorage class, your attributes used find errors, those are not set to private. All in all, I can't find many flaws. You have encapsulated the program, added an iterator, made interfaces and you have setters and getters, the right classes are located in the right packages as well. I think what you have done for this workshop is enough to pass it.