Peer Review
of Salam Al Joranis domain model
by Linda Ott Olander (lo222hd)

This model lacks documentation, which should be added in order to aid understanding.

In use case 1, a username and a password could be added as an attribute to the either the Login or the Authentication object. I think it is unnecessary to have both a Login and an Authentication object, since they symbolize the same thing. There should not be any duplication in the domain model. Different noun phrases may represent the same conceptual class or attribute. [1, p233] The authentication/login object could possibly have an "errormessage" attribute to support the secondary scenario of use case 1.

In use case 4, boat details are missing from the model. It could be added as an attribute of the object Boat.
In the model, I think there should be a connection between Boat and Berth, since each boat is assigned to a specific berth. This is needed in use case 8, where moorings are assigned for boats.

In order to better support use cases 11 and 12, the calendar event could have a title and a start and end date.

Think like a map maker
The domain model uses the existing names in the territory [1, p239].

Static vs. Dynamic view, or Requirements vs. Domain model
This domain model is quite good in showing a static view, which complements the requirements that are more dynamic. But the model could be simplified (for instance by making Login and Authentication the same object), and the association between Boat and Berth is missing.

Focus on reality, not software
The model does this well, but it is somewhat oversimplified. It could be made clearer by adding more attributes, which would make it more useful for a software developer.

Good naming (classes and associations)
The naming is good for both classes and associations, they are clear and easy to understand.

Derived attributes/associations
The domain model has no derived attributes or associations (with the symbol "/" before it [1, p258]). The membership fee could be added as a derived attribute.

Correct UML notation

The UML notation is clear and simple, but it is missing a lot of multiplicity values [1, p242]. The many (*) values are there but not the 0..1, 0 or 1. For instance, a Secretary could list one to many boats, so adding a "1" at the other end of the association line would make the model clearer and more useful.

Understandability/Readability
The model is understandable and readable, but adding documentation, associations, multiplicity values and attributes would make it even more so.

As a developer would the model help you and why/why not?
To some extent, but the model would be more helpful with my already mentioned changes.

Do you think a domain expert (for example the Secretary) would understand the model why/why not?
He/she may have some problem with understanding the Authentication object, otherwise the Secretary should understand the model since it reflects her/his real world domain.

What are the strong points of the model, what do you think is really good and why?
It is simple and easy to read with short, understandable names.

What are the weaknesses of the model, what do you think should be changed and why?
As mentioned earlier, the model is missing a lot of information. Adding documentation, associations, multiplicity values and attributes would make it better.

Do you think the model has passed the grade 2 (passing grade) criteria?
No, I think it should be complemented with said information, with those additions it would pass.

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