

Workshop 1

Peer review regarding Markus Alshraydeh groups domain model. ([Link](#))

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

The model is a bit strange and does not have correct UML notation in many places.

Initially it's a bit hard to grasp what it's about, for example there is a class Member with various associations but what is a member a part of? Is it a book club, a community or a boat club? We are thinking there should be some central class representing what the model is about, like in the examples in Layman's book [1, chp 9.15].

Another major problem with this model is that the attributes of the different classes have been mixed up with some "association thinking". For example the secretary class has an attribute "Uses berth booking list". This isn't an attribute. Almost all classes have the same problem, part of the problem description has become attributes which is not correct UML notation for a domain model.

A good example of the correct way to use attributes is the class "Member". Here we have proper attributes without data types which may not always be needed. For example instead of writing "-First name" you could write "+firstName:String" [1, chp 9.16].

There is also a lot of weak dependency arrows, instead of associations, which are not commonly used in domain modeling [2, Dependency]. The names of those dependencies are also hard to understand, e.g. what is "service", what does it mean.

Another example is the Secretary class that has an association "Uses" to the calendar. As a developer some questions would be raised. What does the Secretary "use" the calendar for? You have an attribute in your Secretary class that says "Manage the club calendar", maybe this "manage" part would be a better association to the calendar? Overall it looks like you have mixed up the meanings with association/lines/arrows [1, chp 9.14]

There are a lot of classes in this model that aren't real-world classes, for example "Register Boat", "Remove Boat" and "Handle Calendar Event". These are more like event or functions even.

We guess the model could be of some help but at the same time we also suspect it would cause a lot of confusion for us as developers.

Do you think a domain expert (for example the Secretary) would understand the model why/why not?

No because there simply too many important classes missing and the associations between the existing classes are very unclear. One example of this is that the Member class has a dependency to Boat Management as stated above which indicates that the member could use some kind of services that the class Boat Management offers, but there is no class for a Boat and therefore no relationship between a member and a boat.

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

The classes that are relevant have good and clear names. The use of a good UML software tool make the model look professional and easy to read.

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

The model lacks a lot of important classes and relationships. The use of attributes is not correct. The model includes software functions which shouldn't be included at all. Most of the weaknesses have been pointed out in the previous sections of this review.

Do you think the model has passed the grade 2 (passing grade) criteria?

No, we don't think this model is sufficient. It should be worked on a little bit more.

- References:**
1. Larman C. Applying UML and Patterns 3rd Ed, 2005, ISBN: 0131489062
 2. https://vaughnvernon.co/?page_id=31