Workshop 1 - Peer Review 1

Sep 16, 2015

Peer review regarding Daniel Nilssons groups domain model. ([Link](https://github.com/danielnilsson9/1DV607))

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

Yes, it would. The model describes both the real-world classes and relationships as well as some parts of a future system (for example the user classes/relationships). The real-world is well described and it is probably possible for a non-expert to understand the situation and problems to be solved. However, when adding parts that belongs to a future software in this stage, there is a risk that wrong decisions are being made that affect the future system negatively.

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

Perhaps the model is a bit too software oriented to be easily understood by a domain model. While real-world concepts such as Member, Boat and Berth are shown with their real world relationships some software specific concepts are also shown which makes the model a bit hard to understand.

For example: \* What is a User and a UserRegister in the real world? \* How is a User different from a Member? \* What is a CalendarEventList in the real world, is it not in fact part of the calendar?

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

It is easy understandable. Use of attributes are excellent. The model use good UML notation. The model contains good relationships between different roles (member, secretary). Associations are clearly shown with relationships and not attributes.

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

It is more suitable for developers then for domain experts. For example, the domain expert could have some problems to understand what is the user’s role in the model. The part about payment is unclear. There is an attribute “cost” for BerthDescriptor, but it is not clear how the membership is determinated. Instead of BoatSize, maybe the better option would be to use BoatDescription and the size as an attribute of BoatDescription. As stated by Larman (1, p.160), most attributes types should be of as primitive data types, such as numbers and booleans. The type BoatType as an attribute of Boat object is a complex domain concept.

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

Yes, it has! Maybe something about updating the membership fee should be added to the model (2. Use Case 4.6 Register boat).

References:

1. Larman C. Applying UML and Patterns 3rd Ed, 2005, ISBN: 0131489062
2. Ohlsson T. Problem description. Workshop 1. Available at: https://coursepress.lnu.se/kurs/objektorienterad-analys-och-design-med-uml/workshops-2/workshop-1-domain-modeling/problem-description/. (Accessed: 2015-09-16)