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

Member-part:  
As a developer the diagram tells me a few rules, a member can own several boats, he can have several payments in his payment history. It is unclear how the author thought when he design the relationship between member and membership registry. All in all, the design is unclear and vague.

Calendar-part:  
The calendar contains events but everything else is left to the developer. We would like to see how the calendar connects to the member.

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

A secretary would be able to understand the model. Since the secretary works at the boat club he can recognize the connections between members and boats etc.

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

The model is easy to understand since it is not very detailed. The overall thought is easy to understand.

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

The domain model should be simple but this model is too simple in our opinion.   
We feel like some important parts of the problem description has been left out in this model which makes it hard to understand the system. The model could use a bit more detail to make system requirements more clear.

**Do you think the model has passed the grade 3 critera?**

We think that if the model is updated so that all requirements are fulfilled and a bit more detail is added, it should pass grade 3.

/

Jesper Holmström  
Eleonor Lagerkrants  
Marcus Darvelid