



The domain model is clear and helpful with its simplicity, both for a developers point of view and the domain experts point of view. With good naming of classes and associations it allows the viewer to understand the domain model.

The conceptual classes within this domain model is strong and not a single one is vague. Together they cover the "loose" requirements within the domain, which follows Larman's guideline of "Think like a map maker" [1. Ch 9.10]. Personally I like the attributes, they complete the conceptual classes which helps the reader to understand the domain a little more. So does the associations, good named and relevant in this model.

Although the model is clean and not cluttered with unnecessary information, it would be good if there had been some more associations. In this kind of small domain model it would not harm, even though Larman states that you should be careful with too many [1. Ch 9.14]. What I think of is some associations between the secretary and the berths, or in a way which

will cover the user case “8 Assign Berths”, since in the domain model secretary is associated with Calendar which is good and cover the user case “10 Manage Calendar Events”.

The domain model is good and cover almost all of the user cases and the contradictory requirements. I would recommend to complete the model with some more associations in order to pass the grade 2 (passing grade) criteria.

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