Sverker Söderlund ss223ck

Review

To ej222ru

The application works good. It’s easy to understand the instructions given in the projects GitHub wiki.

In the sequence diagrams I would like to see the values the methods are returning. It makes it easier to understand what is happening in the methods and why methods are called. “The elements shown in SSDs (operation name, parameters, return data) are terse. These may need proper explanation so that during design it is clear what is coming in and going out.” Larman C. capter 10.8.

The diagrams helps understand the system. You can easily understand the components and how they interact.

Architecture

There is a clear model-view architecture. I would say the model BoatBLL is specialized when using enumerations for input values. If the value is supposed to be constant it should be specified in the database and retrieved from there. Now it’s a dependence between the model and view.

Code quality

The method should have pascal casing.

Design quality

I would like another class that handles business logic instead of the controller. The model should handle domain logic. For example: the saveMember method in Controller is pure business logic and should be moved to a model class.

The control class switch case is dependent on numeric values returning from view. It is not encapsulated and could cause problems if you were to change the code in either the view or the controller.

Generally the design is good and follows object oriented design.

Sverker Söderlund ss223ck

Strong parts

I think the implementation is good. It’s easy to understand the UI. There are many abstract solutions in the code.

Weakness

The biggest weakness is the lack of proper BLL to handle domain logic. The controller is too big and have to much responsibility.

Refrences:

Larman C., Applying UML and Patterns 3rd Ed, 2005, ISBN: 0-13-148906-2