Peer review

Does the implementation and diagrams conform (do they show the same thing)? Are there any missing relations? Relations in the wrong direction?

Good looking diagram and it seems correct with all the relations between the classes. One note that the observer class and relations doesn´t seem to be implemented.

Is the dependency between controller and view handled? How? Good? Bad?

We think that the dependency is removed because the controller is no longer dependent to the simple view according to Larman in applying UML and patterns. (1, Larman)

Is the Strategy Pattern used correctly for the rule variant Soft17?

The soft 17 rule doesnt seem to be correctly implemented according to the rule.

" Soft 17 means that the dealer has 17 but in a combination of Ace and 6 (for example Ace, two, two, two). This means that the Dealern can get another card valued at 10 but still have 17 as the value of the ace is reduced to 1. Using the soft 17 rule the dealer should take another card (compared to the original rule when the dealer only takes cards on a score of 16 or lower)" Workshop 3 instructions.

Is the Strategy Pattern used correctly for the variations of who wins the game?

Yes, with the interface IWinRule() and the class playerWinRule()

Is the duplicate code removed from everywhere and put in a place that does not add any dependencies (What class already knows about cards and the deck)? Are interfaces updated to reflect the change?

In the dealer class you have implemented the function deal and it removes some of the duplication. The dealer is in this case the information expert but the newGameStrategy classes still have the same duplicates, could use the deal function here aswell to shorten down the code by a fair amount, easier to read aswell.

Is the Observer Pattern correctly implemented?

The observer pattern is implemented and seems to work correctly with the notify method.

Do you think the design/implementation has passed the grade 2 criteria?

After some small fixes you will definitely pass for grade 2.

References

1. Applying UML and patterns, Craig Larman,
   ”Dependencies can exist between any elements, but they are probably most often used in UML package diagrams to illustrate package dependencies (see Figure 38.3)”