

Try to compile/use the source code provided. Can you get it up and running? Is anything problematic?

There was no problem to get it up and running.

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

Implementation and diagrams show the same thing, easy to get good overview of the application. Would be good to add details about the packages.

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

yes they are handled in a good way and the hidden dependencies are removed.

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

Yes the strategy pattern is used correctly and desired pattern can be changed in the Rules Factory class. According to journaldev strategy pattern is used for to have multiple algorithms for specific tasks.

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

Same here strategy pattern is used correctly. but only 1 variations is made, another one should be added.

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?

No duplicated code can be found. Interfaces are also updated to reflect changes. Observer interfaces should be moved from the controller to the model.

Is the Observer Pattern correctly implemented?

Yes the observer pattern works well, According to oodesign to have a good design means to decouple as much as possible and to reduce the dependencies. I believe this is true about the project.

Is the class diagram updated to reflect the changes?

Yes.

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

Yes it's very well made and I believe it has passed grade 2.

Referenser

<http://www.journaldev.com/1754/strategy-design-pattern-in-java-example-tutorial>

<http://www.oodeesign.com/observer-pattern.html>