

Peer Review Workshop 3, 1dv607

Fredrik Olsson(folep02) & Andras Ballas(ab223sq)

Reviews:

Chen Ningrui (nc222cv)

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

Even with the our lack in C# experience it went well.

Test the runnable version of the application in a realistic way. Note any problems/bugs.

We have found a one problem with the rules:

Even if the player goes bust the dealer continues playing, but the dealer still wins if the the dealer gets a higher score than the player.

Maybe it's out of the scope for the peer review but we would prefer if the the whole view was rendered at each deal instead of a cleared console and the card. It's really not a big thing but it feels kind of pointless because you to memorize the already given cards to feel the "excitement" the pause adds to the game.

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

Yes the implementation and diagram seems to conform

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

Yes changed the letters used for playing the game in the view it still works.

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

There doesn't seem to be an implementation of the Soft 17 rule.

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

No we can't find any rules at all under rules for deciding the winner.

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?

We don't see how the code duplication is removed. Maybe we just don't understand the implementation of the CardDataProvider.

Is the Observer Pattern correctly implemented?

We have problem figuring the out the implementation and think that maybe it would be better to add the controller as an observer instead of printing to the console directly from the update method in the observer, also the thread sleep happens in the update method but according to the requirements for the workshop it should be handled in the UI.

Is the class diagram updated to reflect the changes?

The class diagram seems to be correct.

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

We think that the application needs some fixes for passing grade 2.