

Lisa Westlund,
Caroline Forslund,
Johanna Szepanski

Peer review av Rebecca Fransson, Hampus Jarleborn, Richard Söderman WS 3.

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

Inga problem. Det fungerar även när vi väljer att implementera olika regler.

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

Korten listas innan de tilldelas rätt spelarhand. Ni kallar på metoden DisplayCard() i controllerns metod CardDrawn(). Återanvänd istället koden på rad 23-25 i controllern.

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

Ja, de har använt enums vilket känns som rätt val.

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

Det är rätt implementerat.

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

Det borde ha skapats två olika strategier beroende på vem som ska vinna, en dealerWins och en playerWins. Som det är nu finns bara en PlayerWinsOnEqual. Tänk som ni gör med SoftSeventeen i RulesFactory där ni returnerar olika strategier beroende på vilken regel som ska gälla. Precis likadant ska ni göra i rulesfactory:GetWinnerRule() returnera olika strategier beroende på vilken vinnarregel som ska gälla. Så som ni gör nu får ni en onödig dependency med dealer. Se Larman [p447, ch26.7].

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?

Koden är refaktoriserad och utflyttad till en metod i Dealer och interfaces är uppdaterade. Det finns dock en icke använd dependency till Deck. Se INewGameStrategy och de klasser som implementerat interfacet.

Is the Observer Pattern correctly implemented?

Vad vi kan se är det rätt implementerat. Enda kommentaren hänvisar vi till frågan ovan under "Note problems/bugs".

Is the class diagram updated to reflect the changes?

Till viss del är det uppdaterat, men se kommentarer nedan.

PlayGame

Associationernas rollnamn ska inte vara klassernas namn utan namnen på attributen och de ska placeras vid pilen. Se Larman [p522, ch 31.13] och de associationer som fanns i klassdiagrammet från början.

Dependency till Card saknas såsom koden är skriven i nuläget, men när ni rättar till buggen enligt ovan kommer dependency från CardDrawn() till Card att försvinna.

Dealer

Associationen från Dealer till IWiningStrategy bör förändras på samma sätt som associationerna från PlayGame. Detsamma gäller associationen till IObserver.

Dealer ska ha en association till RulesFactory. Nu är det en dependency.

Dealer bör även ha en dependency till Card.

Game

Har dependency till RulesFactory som inte är med.

IObserver

Ska ha ett dependency till Card.

Vi ser att klassdiagrammet inte visar dependencies från Rules->Interfaces till modelklasser. Detta är helt ok då det gör klassdiagrammet mer lättläst, men ni bör skriva en kommentar om att ni valt bort dem och varför. Detsamma gäller även att RulesFactory enbart har dependencies till strategiklasserna och inte till interfacen i ert diagram.

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

Ja, bara ni fixar implementationen av WinnerStrategy, buggen med att korten delas ut innan de tilldelas rätt spelaren och uppdaterar klassdiagrammet.

Referenser

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