Peer Review

Compile Test: We could not find a runnable jar file for your project. We went to the read me file to see if there is any instructions for running the black jack game. Since we could not find any instructions we decided to evaluate your work given the circumstances.

Documentation and Implementation consistency

1. You specified that interface iView has a dependency to the Card class, rather the interface specifies methods for classes implementing iView.
2. You specified that the controller PlayGame has a dependency to the iView, rather it is an association.
3. You specified that the controller PlayGame has a dependency to the class Game, rather it is an association.
4. We think that the Player should have a dependency to the RoolsFactory instead of a dependency from Dealer to the RoolsFactory, since the Dealer inherits the Player.
5. We are missing the classes Soft17Strategy and TieStrategy in your class diagram.

Realistic Game Test: See Compile Test.

Controller View Dependency Solution: Task failed, you have not completed the task to get rid of the string dependency in the switch that delegate user input.

Strategy Pattern: Correctly implemented.

Strategy Pattern (win rools): Task failed, you have not created the iWinStrategy interface, and neither have you created the strategy class that implements the interface.

Duplicate Code: You somehow reduced the duplicated code in InternationalGameStrategy and AmericanGameStrategy, but in AmericanGameStrategy you accidently created a bug since last card dealt by dealer should be hidden.

Observer Pattern: As far as we can see, there is no implementation of the observer pattern. Maybe some code is missing? The controller class PlayGame implements an Observer that we can't find in your repository.

Summary: Your workshop is incomplete, you need to implement the failed tasks and update your class diagram to reflect your code result.