Euchre

Milestone 3

Logan Hallowell
Alex Memering
Erik Sanders

Table of Contents

1 Revised Project Plan

1.1 Key Business Features

Currently we are on track with our previous project plan. The first milestone went smoothly, and we accomplished all of our goals. For this milestone, we have the AI players correctly playing their cards in a greedy manner which follows the many complicated rules that are associated with trump and what was lead. We also have the scoring system for tricks finished; each pair has a score that is listed after each card is played and updates correctly. This now allows the user to play through almost all of a round. Our next major feature to implement involves playing from the beginning of a round through to its completion.

1. Milestone 4
    a. Score rounds
    b. Select Trump
    c. Dealer shuffles cards and deals them out correctly
    d. Deal at the beginning to determine first dealer
2. Milestone 5
    a. Going alone

2 Code Coverage Report

We currently have 80.2% code coverage according to our code coverage tool, EclEmma. However, this number is low because the tool includes classes that we do not test, such as ApplicationWindow and Controller. Our code coverage without these classes is roughly 97%. We need to revise our code tests for Card, so that we actually test the toString and equals methods instead of testing them through other class' tests. We should also look through other classes to ensure we did not repeat this oversight.

3 Class Diagram

Card

+suit: SUIT
+value: int

+Card(SUIT s, int val): Card
+toString(): String
+equals(Object obj): boolean
+cardValue(SUIT trump, SUIT leading): int
+greater(Card other, SUIT trump, SUIT leading): boolean

0..24
deck

0..5
hand

Player

+name: String

+Player(String name): Player

1

4 players

currentWinningCard

0..4
cardsPlayed

1   1

AIPlayer

+AIPlayer(String name): AIPlayer
+getCardToPlay(Trick trick): Card

Trick

+leadingPlayer: int
+leadingSuit: Card.SUIT
+currentPlayer: int
+currentWinner: int
+trump: Card.SUIT

+Trick(int leadPlayer, Card.SUIT tr): Trick
+playCardForCurrentPlayer(Card c): void
+isStarted(): boolean
+isOver(): boolean
+getNextTrick(): Trick

1 currentTrick

Euchre

+currentTrick: Trick
+trump: Card.SUIT

+Euchre(): Euchre
+shuffle(): void
+draw(): Card
+deal(): void
+humanPlayCard(String cardBeingPlayed): void
+isCurrentPlayerAI(): boolean
+makeAIPlay(): void

1

1

1

1

euchre

1

Controller

+getAllHands(): ArrayList<ArrayList<Card»
+cardPlayed(String cardText): void
+getTextForMiddle(): String
+setAppWindow(ApplicationWindow appWin): void
+stringOFTrickPlayed(Trick trick): String

ApplicationWindow

+ApplicationWindow(Controller cont): ApplicationWindow
+main(String[] args): void
+setMiddleTextArea(String textForMiddle): void
+updateHands(ArrayList<ArrayList<Card» currentHands): void