Grid Answers

March 27, 2016

6:18 PM

**Game Logic (to be tested without networking)**

*Each Tournament Testing*:

1. one player draws/starts, others draw but do not participate (ie withdraw)

***Test:*** class TestGameEngine: testPlayerStarts

1. one player draws/starts, others draw but only one participates by playing a card or several cards

***Test*:** class TestGameEngine: testPlayersParticipate

1. one player draws/starts, others draw and some participate by playing a card or several cards

***Test:*** class TestGameStart2Player

1. one player draws/starts, other draw and all participate

***Test:*** class TestGameStart2Player

1. starting with a supporter or several supporters

***Test:*** class TestGameEngine: testStartingWithSupporters

1. a multiplayer tournament has several rounds where each player plays one and then several supports in different rounds

***Test*:** class TestGameEngine: testMultiplayerTournamentWithSupporters

1. trying to play cards that do not get the current player to beat the tournament originator (ie not enough to be the leader)

***Test*:** class TestGameEngine: testTotalValueCards

1. restriction to 1 maiden per player per tournament

***Test:*** class TestGameEngine: testMaiden

1. winning and getting a token

***Test:*** class TestGameEngine: testWinnerToken

1. winning and choose token when purple tournament

***Test:*** class TestGameEngine: testWonPurpleTournament

1. losing with a maiden and losing a token

***Test:*** class TestGameEngine: testLoseOnMaiden

*Each Action Card Testing*:

1. playing this card on an unshielded player

***Test:*** class TestActionCards:

TestUnhorse

TestChangeWeapon

TestDropWeapon

TestBreaklance

TestRiposte

TestDodge

TestKnockDown

TestOutmaneuver

TestCharge

TestCountercharge

TestDisgrace

TestAdapt

TestAdapt

TestOutwit

TestShield

TestStunned

TestIvanhoe

1. playing this card on an shield player

***Test:*** class TestActionCards:

TestUnhorseShielded

TestChangeWeaponShielded

TestDropWeaponShielded

TestBreaklanceShielded

TestRiposteShielded

TestDodgeShielded

TestKnockDownShielded

TestOutmaneuverShielded

TestChargeShielded

TestCounterchargeShielded

TestDisgraceShielded

TestAdaptShielded

TestAdaptShielded

TestOutwitShielded

TestShieldShielded

TestStunnedShielded

TestIvanhoeShielded

1. undoing this card using Ivanhoe

***Test***: class TestActionCards:

TestActionCards:

TestIvanhoeUnhorse

TestIvanhoeChangeWeapon

TestIvanhoeDropWeapon

TestIvanhoeBreaklance

TestIvanhoeRiposte

TestIvanhoeDodge

TestIvanhoeKnockDown

TestIvanhoeOutmaneuver

TestIvanhoeCharge

TestIvanhoeCountercharge

TestIvanhoeDisgrace

TestIvanhoeAdapt

TestIvanhoeAdapt

TestIvanhoeOutwit

TestIvanhoeShield

TestIvanhoeStunned

1. checking a used action card is indeed thrown away

***Test:*** class TestActionCards: all tests in this class

*Scenario Testing*:

1. the player who start cannot start a tournament

***Test:*** class TestScenarios, testPlayerCannotStart

1. last tournament was purple, cannot be purple again

***Test:*** This is handled on client side by not offering purple as an option, and cannot be checked in the game engine.

1. trying to play an insufficient number of cards to become the leader on my turn

***Test*:** class *TestScenarios*, testTotalValueCards

1. trying to play invalid cards

***Test:*** *class TestScenarios, testInvalidCards*

1. coming to end of the deck

***Test:*** *class TestScenarios,* testEndOfDeck

1. using 'Charge' in a green tournament with every player with only green 1s: one card must remain

***Test:*** *class TestScenarios,* testChargeGreenOneCard

1. other example of overriding rule: at least one card must remain

***Test:*** *class TestScenarios, testOneCardRemains*

1. winning the game

***Test:*** class *TestScenarios*: testWinnerToken

1. the deck uses 110 cards

***Test:*** class *TestScenarios*, testNumCards