

Acceptance Test Procedures

Brandon Smith, Nieka Gutenberger, Joseph Coppin, Ryan Frazier, Trevor Jewkes

October 22, 2016

1 Server

1.1 Server

Validation:

- Client can connect to the server when opening the client.

Verification:

Unit Test for connecting to server.

2 Client

2.1 Login

2.1.1 Create user

Validation:

- Create a new user to 'log in' to access server.
- Create username to identify player.
- Create password.
- Ability to change password if needed
- Successfully login or logout each game.

Verification:

Unit Tests for creating user functions.

- unit test for function createAccount()
- unit test for function userSettings()

2.2 Main Menu

2.2.1 Main menu

Validation:

- Menu to edit account/profile
- Choose specific game
 - Choose Hearts
 - Choose Spades
- Option to join game or create game
 - In joining game, able to select/specify table to join
 - In creating game, choose options for gameplay
 - * Choose Computer Players or AI

* Choose Private or Public

Verification:

Unit Test for choosing game, ensuring only specific options work when creating/joining games.

- unit test for function chooseGame()
- unit test for function gameOptions()

2.3 Game Table/GUI

2.3.1 Cards

Validation:

- View cards, ensure cards are dealt properly
- Cards match values listed on card
- See how many cards other users have (face down)

Verification:

Unit Tests for how cards will look, how cards are shuffled, how many cards are visible to player.

- unit test for function createTable()
- unit test for function showHand()
- unit test for function drawCard()

2.3.2 Scoreboard

Validation:

- See scores/bid
- Reflects appropriate score at the end of each turn
- Players info and cards sent each time a player selects a card.

Verification:

Unit Tests for ensuring correct information is passed to scoreboard.

- unit test for function displayCenter()

2.3.3 Moves

Validation:

- See when other players move
- Each turn is shown with appropriate amount of time
- Can see past hand from option menu

Verification:

Unit Test for ensuring correct information is sent to display.

- unit test for function `redrawTable()`
- unit test for function `getHint()`
- unit test for function `viewHistory()`

2.4 Game End Event

Validation:

- If connection removed prematurely, AI will fill in so gameplay will continue without problems
- End at proper time
- Score for round displays on screen
- Option to play again
- Option to return to main menu

Verification:

Unit Test for making sure all premature exits are taken care of by creating and using AI.
Unit Test for making sure each function dealing with new game works correctly.

- unit test for function `endOfGameDisplay()`

2.5 Other

2.5.1 Multiplayer Options

Validation:

- AI fills in as needed
- 1-3 other humans can join game
- Disconnecting user is replaced with AI that takes place
- After determined amount of time of inactivity, player is removed from game
- When disconnected accidentally, can rejoin same game if game is still in play

Verification:

Unit Test for ensuring that multiplayer functions work as expected.