EuchreIQ Test Plan and Results

## Overall Test Plan

Our test plan will include two aspects of our project, a run through of the game and a functionality test from an admin’s perspective. The run through will be a simulated experience of a player interacting with the game. We will create test cases that should lead to expected actions of the game for the user. These can include but not limited to calling trump, playing a card, and winning or losing a hand. The functionality test from the admin’s perspective will be done through checking specific areas of the program related to aspects such as data transfer, creating new games, and editing data for the bots.

## Test Case Descriptions

1. **Mechanics Test 1**
2. Ensure cards are dealt with randomness
3. Verify that the deck is shuffled before each round and each player receives an equal number of unique cards
4. Inputs: Start of new game
5. Outputs: Each player has exactly 5 unique cards
6. Normal
7. Blackbox
8. Functional
9. Unit
10. **Mechanics Test 2**
11. Ensures that only valid cards can be played
12. This ensures that each player can play a valid card that doesn’t violate the rules of Euchre (playing a card that is off-suit of the card played).
13. Inputs: Attempt to play a card that is off-suit of the card that was led.
14. Output: The game prevents the card from being played and requires the user to play a valid card
15. Abnormal
16. Blackbox
17. Functional
18. Unit
19. **Mechanics Test 3**
20. Ensure correct trick winner
21. Verifies at the end of each trick, the team with the greatest card played wins the trick.
22. Inputs: Play a standard round of Euchre
23. Outputs: The trick is awarded to the card of the highest value.
24. Normal
25. Whitebox
26. Functional
27. Integration
28. **User Interface Test 1**
29. Ensure the UI correctly displays game state changes.
30. Verify that the UI updates player hands, trick winners, and scores dynamically.
31. Inputs: Play a complete game.
32. Outputs: UI reflects accurate game progress.
33. Normal
34. Blackbox
35. Functional
36. Integration
37. **User Interface Test 2**
38. Ensure buttons trigger the correct actions during gameplay.
39. Verify that buttons for card selection, passing, and playing function correctly.
40. Inputs: Click action buttons.
41. Outputs: The appropriate game action is triggered.
42. Normal
43. Blackbox
44. Functional
45. Unit
46. **Admin Test 1**
47. Ensure authorized users can access admin features and information
48. Verify authentication enforces valid login credentials
49. Inputs: Input correct and incorrect login information
50. Outputs: Access is granted for correct login credentials and denied for incorrect login credentials
51. Normal/Abnormal
52. Blackbox
53. Functional
54. Unit
55. **Bot Test 1**
56. Ensure bots are making correct decisions
57. Verify that, given the game state, the bots are determining the most effective play in calling trump and in playing each trick.
58. Inputs: Play a complete game
59. Outputs: Bots make the optimal move
60. Normal
61. Whitebox
62. Functional
63. Integration
64. **Data Test 1**
65. Ensure player stats are saved correctly
66. Verify that after each game, any stats are correctly logged for the player
67. Inputs: Play a complete game and check data after individual moves
68. Outputs: Stats are correctly displayed
69. Normal
70. Blackbox
71. Functional
72. Integration

## Test Case Matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case ID | Normal/Abnormal | Blackbox/Whitebox | Functional/Performance | Unit/Integration |
| MT 1 | Normal | Blackbox | Functional | Unit |
| MT 2 | Abnormal | Blackbox | Functional | Unit |
| MT 3 | Normal | Whitebox | Functional | Integration |
| UI 1 | Normal | Blackbox | Functional | Integration |
| UI 2 | Normal | Blackbox | Functional | Unit |
| AT 1 | Normal/Abnormal | Blackbox | Functional | Unit |
| BT 1 | Normal | Whitebox | Functional | Integration |
| DT 1 | Normal | Blackbox | Functional | Integration |