

Draft a Test Plan

Objective: Ensure the game "Duel Duo" functions correctly in all aspects, including the user interface, game mechanics, and correct calculation of wins and losses.

Scope:

- Functional Testing: Verifying game mechanics, robot drawing, dueling, and win/loss calculation.
- Interface Testing: Assessing the usability of the game interface, especially the "Draw" button and robot selection process.
- Reliability Testing: Ensuring the game consistently resets wins and losses upon page reload.
- Resources: Web browser for playing and testing the game.
- Schedule: Allocate specific time for each test phase, for example, 1-2 hours for initial testing and additional time for re-testing after issues are addressed.
- Test Cases: Include various scenarios covering all functionalities of the game.

Bug Reporting: Document any anomalies or issues encountered during testing.

Define Test Cases

Test Case 1: Draw and Select Robots

- Objective: Ensure that clicking "Draw" displays robots correctly and allows for selection.
- Steps: Click "Draw", select two robots, observe if the computer selects two robots automatically.
- Expected Result: Five robots are displayed, player and computer each select two robots.

Test Case 2: Duel Mechanics and Win/Loss Calculation

- Objective: Verify the duel mechanics and the correct calculation of winner.
- Steps: Engage in a duel, note the health and attack values, verify the winner is correctly calculated.
- Expected Result: The winner should be the one with the higher calculated score.

Test Case 3: Resetting Wins and Losses

- Objective: Confirm that wins and losses reset when the page is reloaded.
- Steps: Play a few games, note the win/loss record, reload the page, check the win/loss record again.
- Expected Result: Wins and losses reset to zero upon page reload.

Bug Report: Incorrect Player Record Update in Duel Logic

Description

- In the /api/duel endpoint of the game server, there is a logic error concerning the update of the playerRecord. Regardless of whether the player wins or loses the duel, the losses counter is incremented. This results in inaccurate tracking of the player's wins and losses. The expected behavior is for the wins counter to increment when the player wins and the losses counter to increment when the player loses.

Steps to Reproduce

- Start a game and proceed to the dueling phase.
- Complete a duel where the player wins (player's total health and attack damage exceed that of the computer's robots).
- Observe the update made to the playerRecord.
- Repeat the duel process, this time ensuring the player loses (computer's total health and attack damage exceed that of the player's robots).
- Again, observe the update made to the playerRecord.

Expected Results

- When the player wins a duel, the wins counter in the playerRecord should increment by 1.
- When the player loses a duel, the losses counter in the playerRecord should increment by 1.

Actual Results

- Regardless of winning or losing the duel, only the losses counter in the playerRecord is incremented. The wins counter does not increment, even when the player wins the duel.

Environment

- Server-side environment running the Duel Duo game.
- Node.js
- Express.js

The issue is observable in the server logic, specifically in the /api/duel endpoint within the server.js file.