Match or Miss – Testing & UX Design Changes

Areas Needing Improvement to Achieve a Pass:

LO1 – Design and Develop a Dynamic Front-End Web Application

1.1 – Application Design

Your design needs to follow accessibility guidelines (e.g., sufficient contrast, alt text for images, keyboard navigation).

Apply UX principles such as intuitive layout, consistency, and user guidance.

Improve your navigation structure so that users can easily find their way through the app.

Ensure your application's design clearly aligns with its intended purpose – for example, better signposting or clear instructions to the user.

Actions:

Add design rationale or UX diagrams to support your implementation in the README.

LO3 – Testing the Application

3.1 – Manual Testing

You did not provide clear manual test procedures to assess your app's functionality, usability, and responsiveness.

3.6 – Test Documentation

You need to document test results, including:

What was tested (feature, browser/device),

Expected vs. actual results,

Any bugs found and how you fixed them (or if left, why).

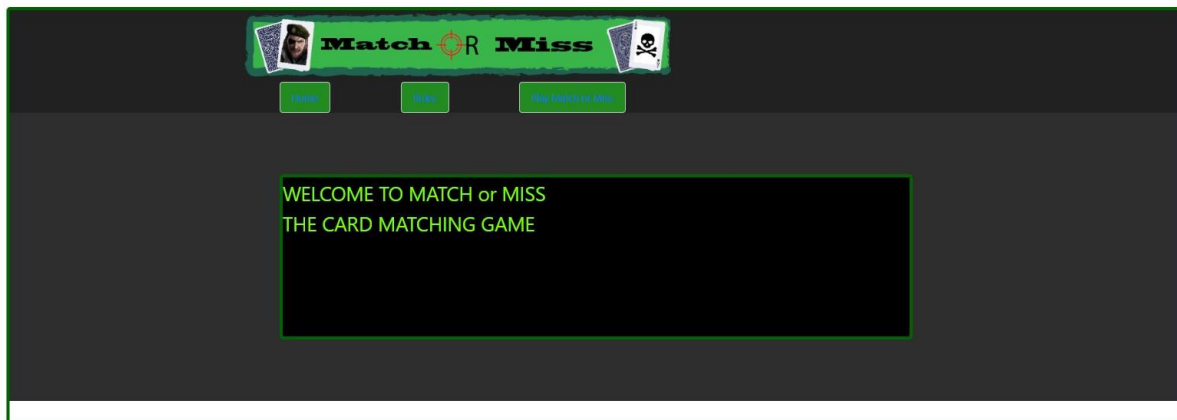**LO1 – Design and Develop a Dynamic Front-End Web Application**

1.1 – Application Design
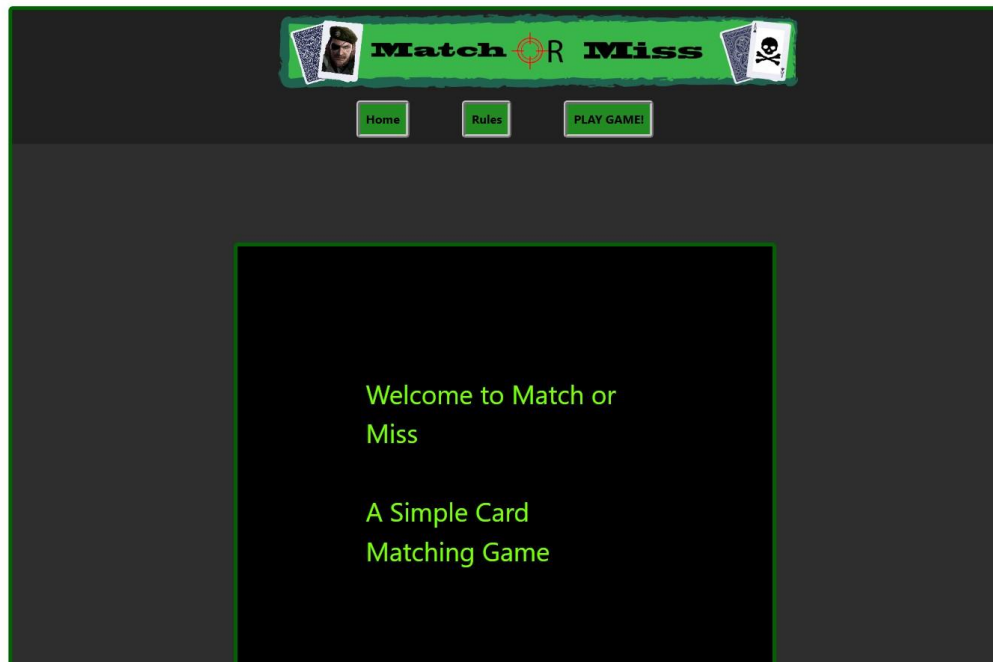
Improved UX and navigation,

Adjusted font colour in navigation bar buttons to black against a green background. This was done to replace the light blue/turquoise font color that was difficult to see against the green background.

The Welcome Page

The "Old" Welcome Page

The New Welcome Page

Made all button elements tabbable, navigation bar, deal button to increase accessibility and to add to users experience.
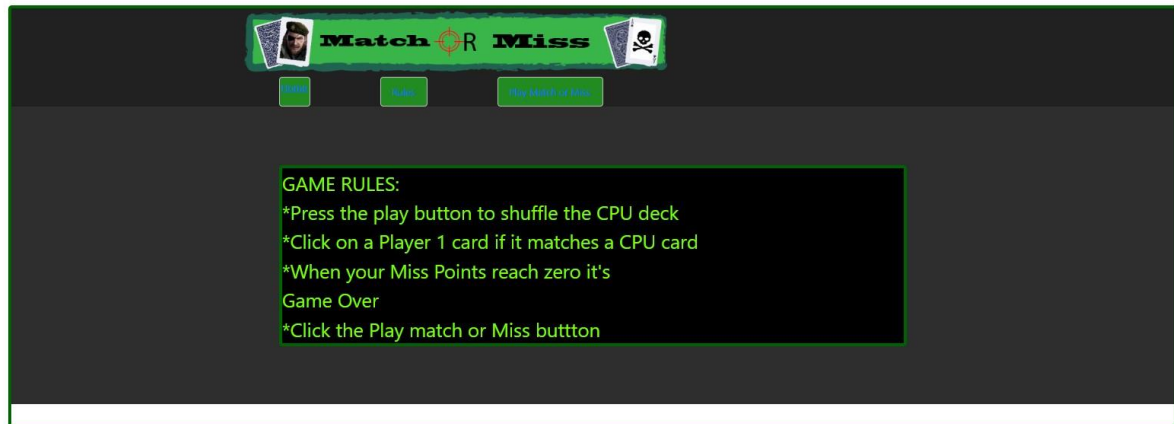
Clearly defined layout and purpose.

Bootstrap Navigation Bar

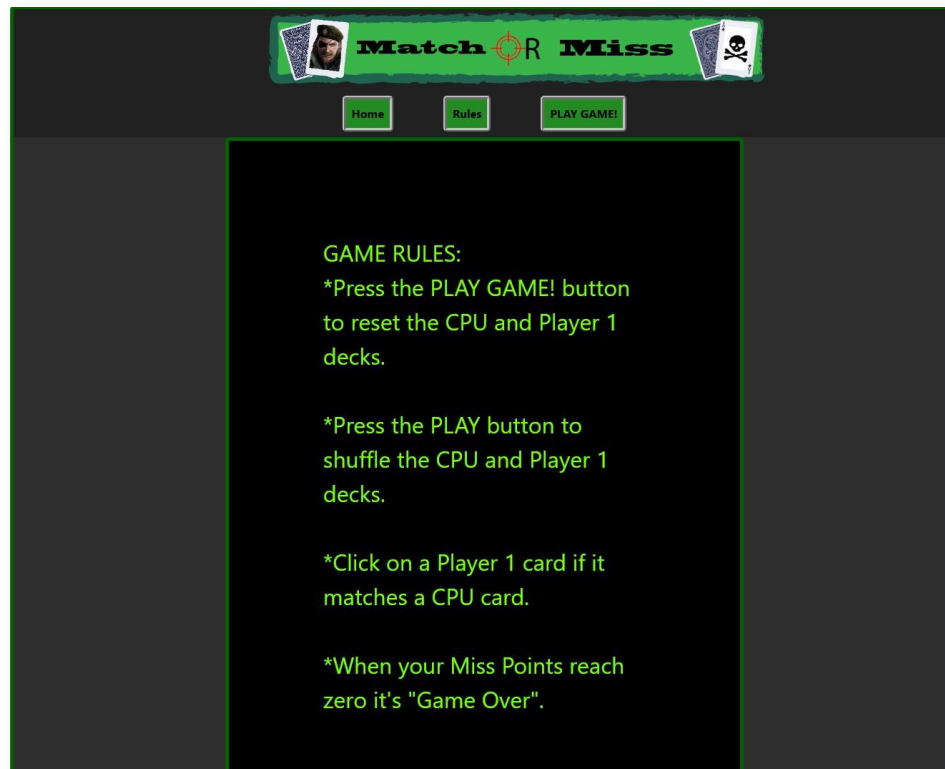I used a bootstrap navigation bar. Replacing the old <a> links made to look like buttons

The Rules Page

The old "Rules Page" lacked styling and failed to give a clear idea to the user as where to go.
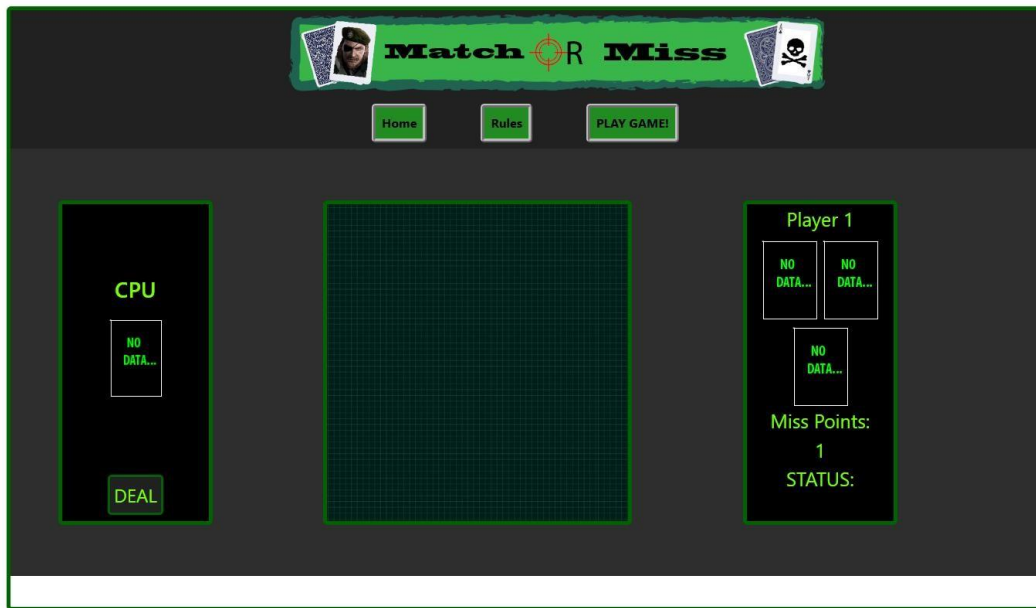
The newly designed "Rules Page".

Match OR Miss

Home | Rules | PLAY GAME!

GAME RULES:
*Press the PLAY GAME! button to reset the CPU and Player 1 decks.

*Press the PLAY button to shuffle the CPU and Player 1 decks.

*Click on a Player 1 card if it matches a CPU card.

*When your Miss Points reach zero it's "Game Over".

The Games Page The "Old" Games Page

The New Games Page

LO3 – Testing the Application

3.1 – Manual Testing

Action:

Create a table or checklist of manual test cases and outcomes.

Manual Test Cases:

Web Application Usage

Game Start = When the "Play Game" button is pressed a 5 second countdown should begin.

Expected:

When "Play Game" button is pressed, the page is loaded and the AI voiced game countdown from 5 to 1 begins.

Actual Results:

The countdown function did experience some issues to begin as variable

Let c = setinterval (showclock,1000) seemed to interfere other areas of code.

After setting the variable as: var c = setinterval (showclock,1000) the countdown function was working normally.

CPUPLAY = A function activated when the "DEAL'' button is pressed.

Expected**:**

Both CPU and Player 1 decks are shuffled and dealt to their respective decks on screen.

All decks randomized successfully most of the time with rare occasions where all player 1's decks drew the same card.

Card Checker: A function that compares a randomly selected card for CPU's deck to a randomly selected card from Player 1's deck

Expected:

After the cardchecker has run the user will be played a sound indicating if they made the right choice or not.

Actual Results:

Cardchecks were being processed with an occasional error incorrect selection happening even though the cards were identical.

HealthDeduct: A function used to deduct health points from player 1 each time they guess incorrectly.

Expected:

1 Point deducted from 10 after each incorrect card is picked

Actual Results:

This function could only go as far removing 1 point from the total of 10. As a result, I set the initial value to 1 allowing the game to finish in theory.

Further Testing

CARDCHECKER:

Ran manual tests on each cardchecker function 1 2 and 3 with correct results when arguments are triggered e.g "random card === random1 card".

SOUNDS:

Tested to ensure correct sounds were triggered if there was a Match or Miss outcome.

The web application was tested across Chrome, firefox and MS Edge successfully.

Bugs:

LOOPS:

I was unbale to create the desired game loop when Player 1 Miss Points reached zero.

Game Over message:

Despite adding the innerHTML code I was unable to make to get the status to display the "Game Over" message.

3.2 - Insert screenshots of the finished project that align to relevant user stories

User Story - One : I used to love playing cards games on my old PC

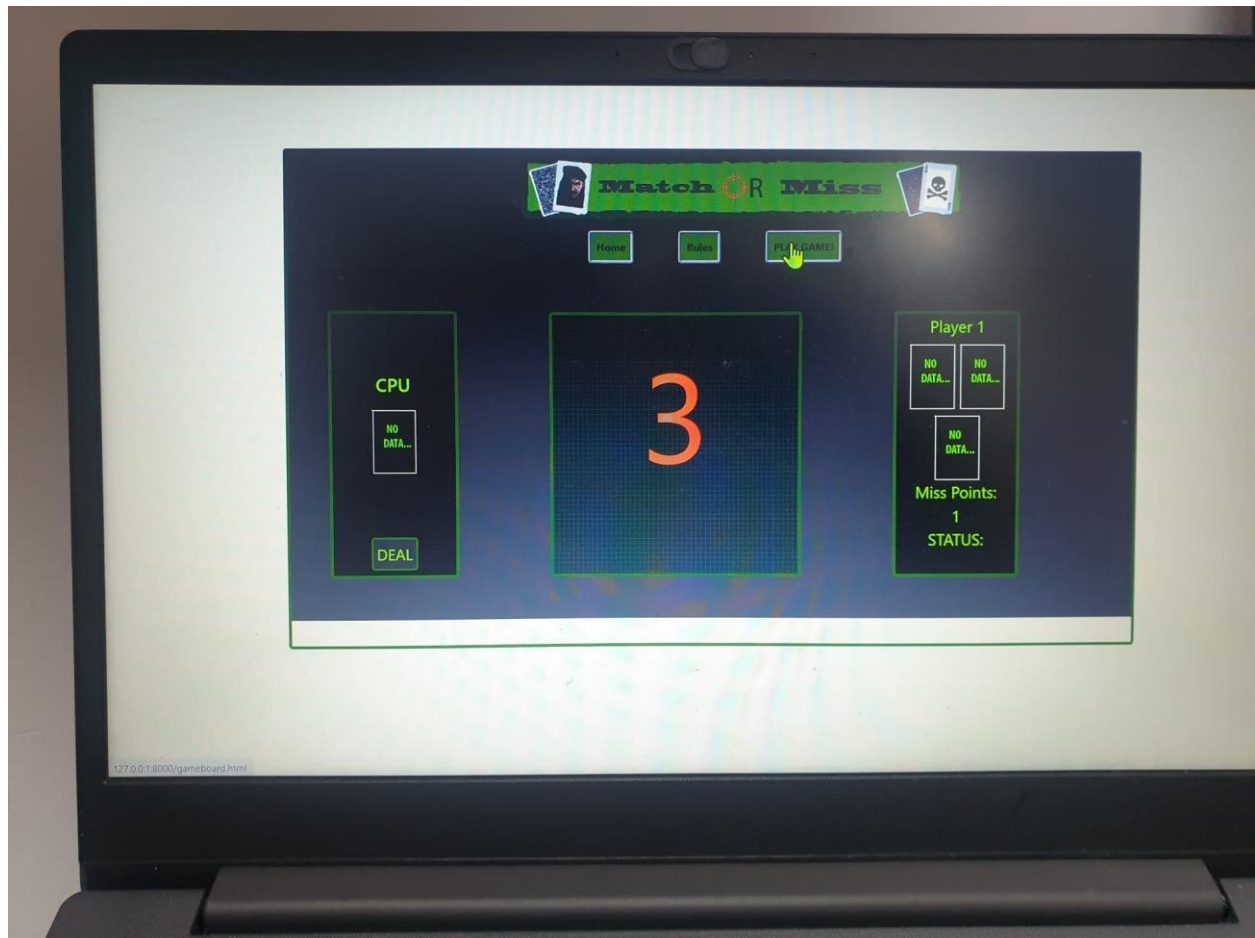"I just love card games and would really love a card game I can play by myself when I'm bored"

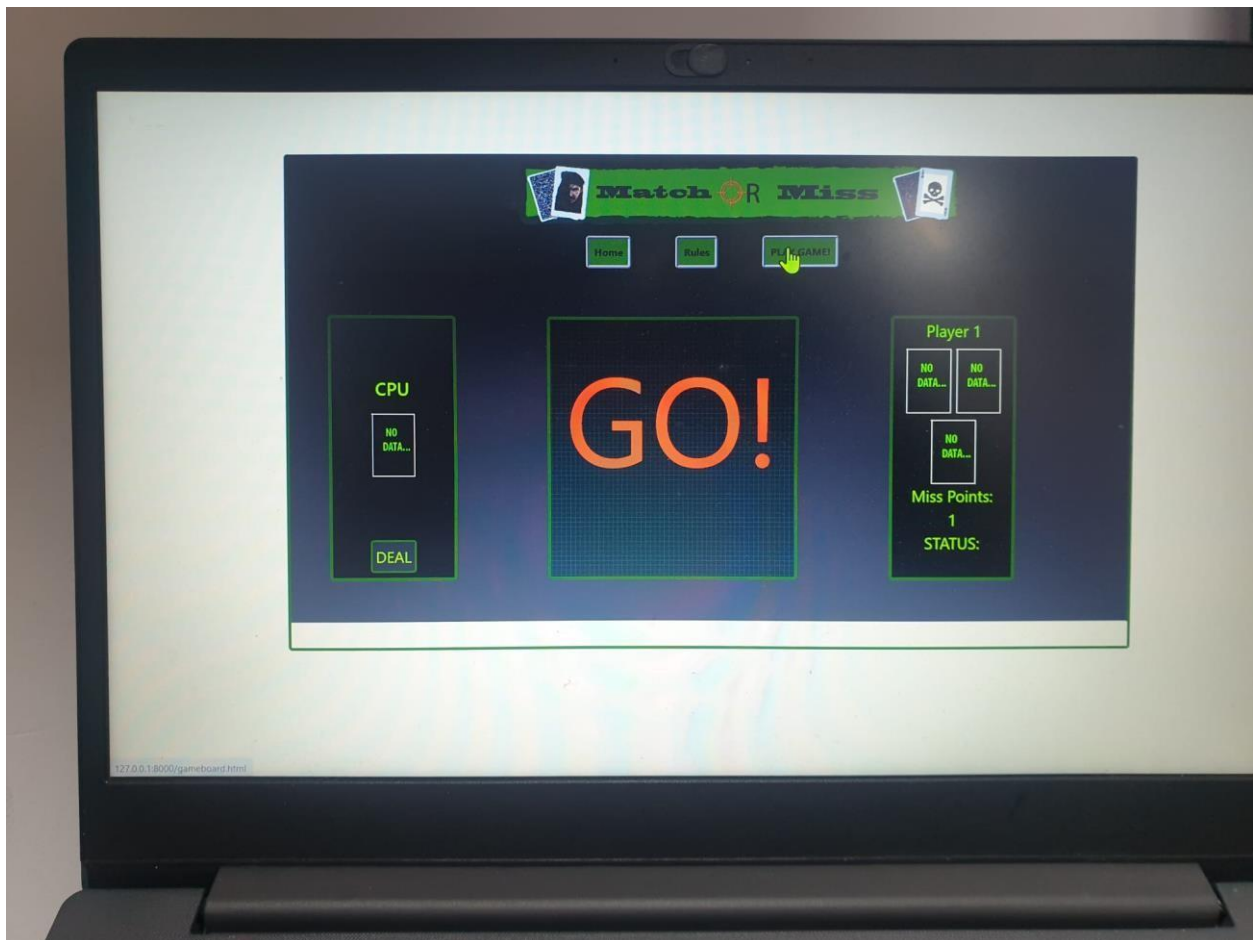**Acceptence Criteria Met:** User can play a card game by himself.

In Game Pictures

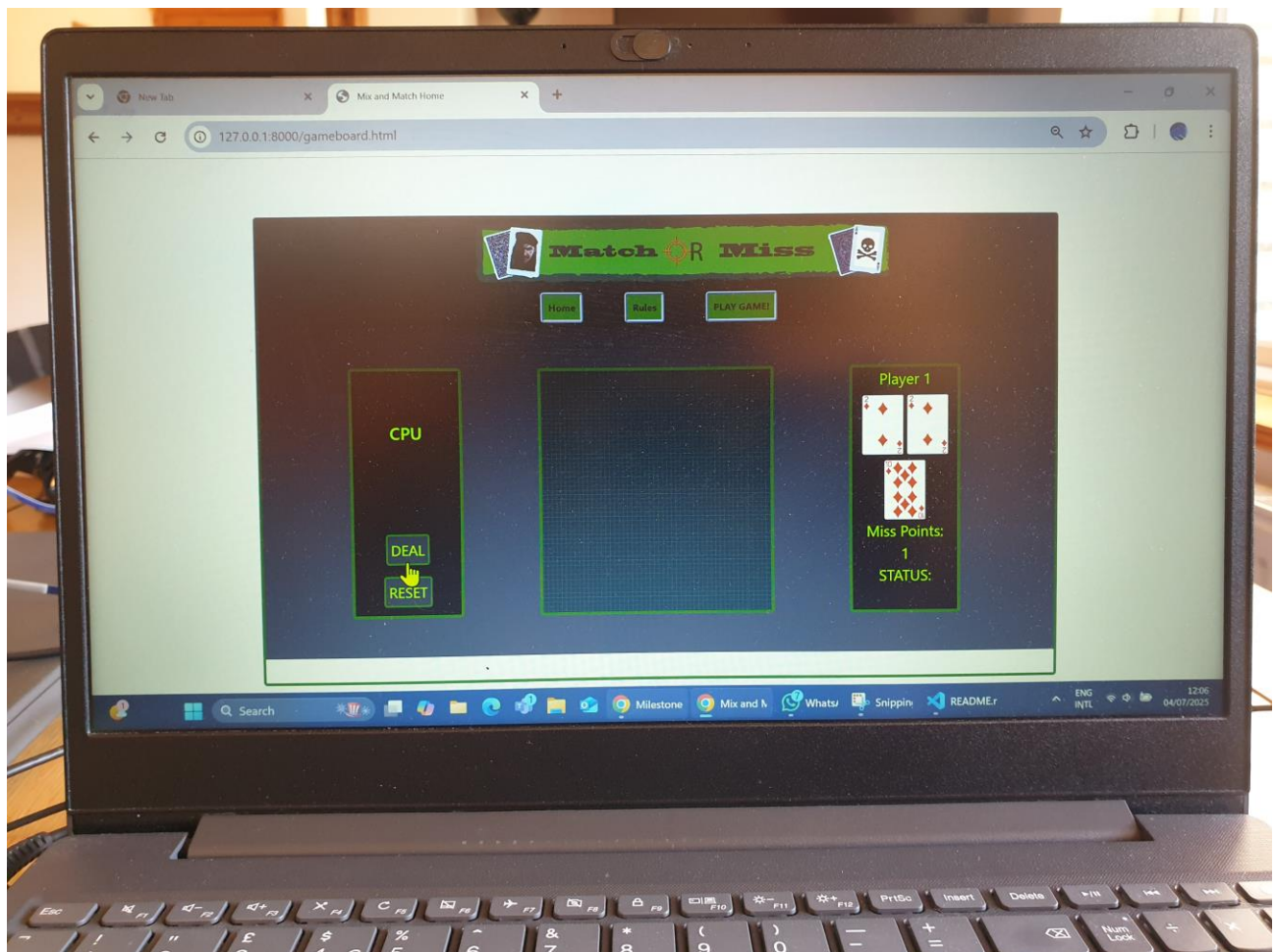**Game Board Ready!** (User approaches the Game Board)

Countdown Begins! (A serious sounding voice gives the user gives as 5 sec countdown to ease the user into the game)
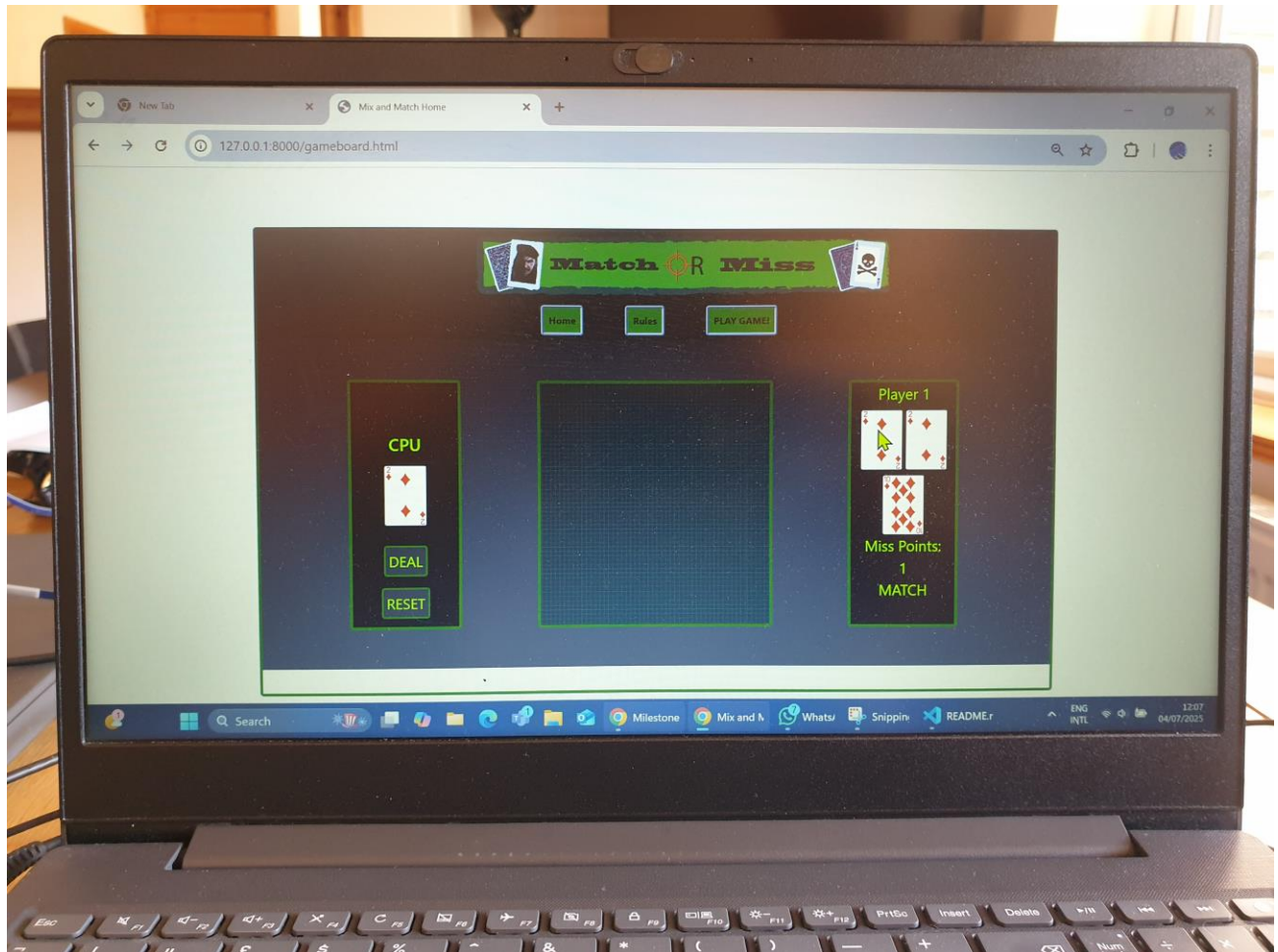
**GO!** (Here is the signal for the user to hit the "Deal" button)

DEAL THE CARDS (User clicks "DEAL" button. Both User (player 1) and CPU decks are shuffled, with CPU card disappearing after 2 seconds)

It's a Match! (User has clicked on and matched their Player 1 card (2 of Diamonds) with the CPU card (2 of Diamonds).

2ⁿᵈ DEAL (User goes for a 2ⁿᵈ deal of the cards, with CPU card disappearing after 2 seconds)
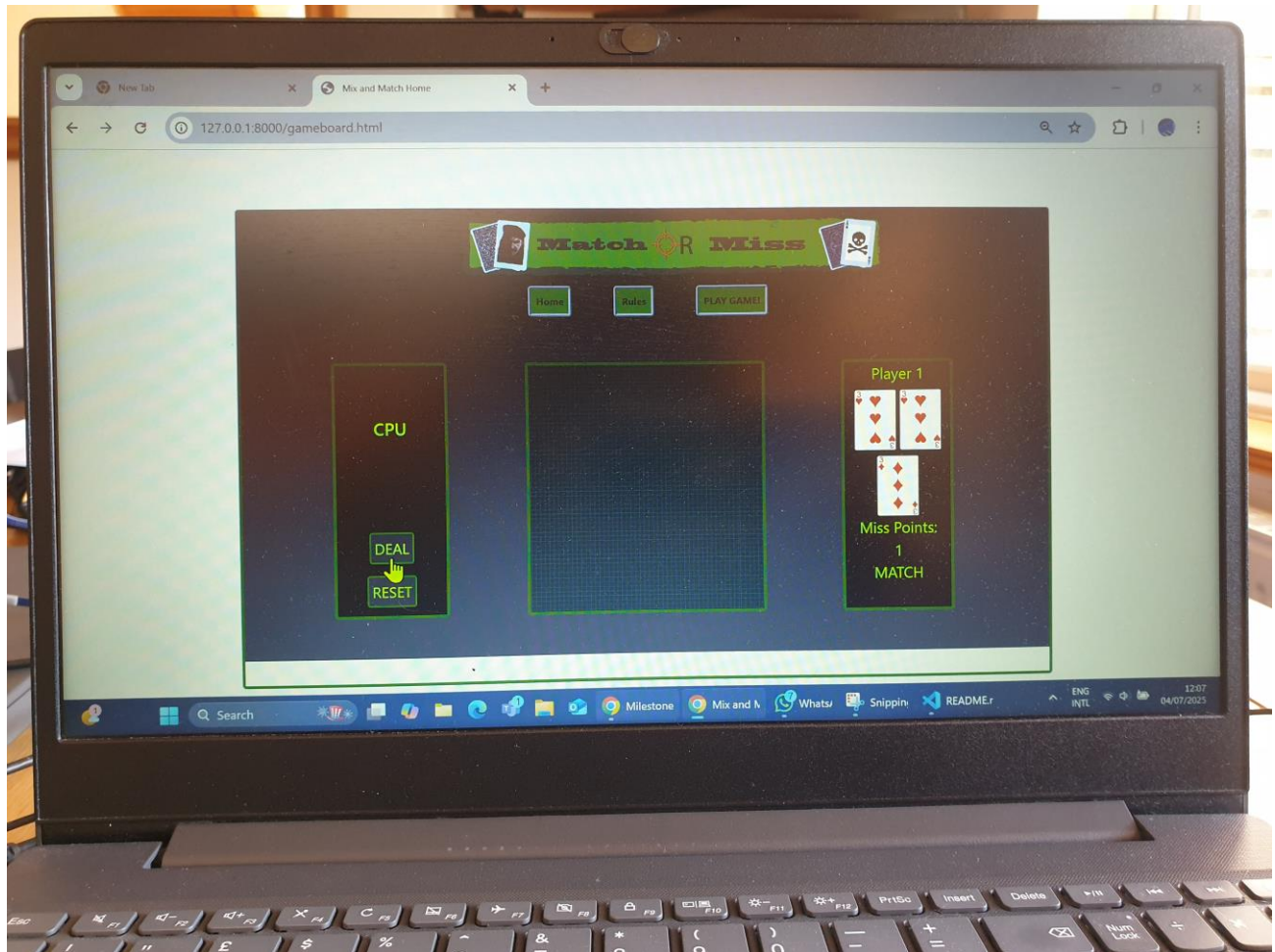
It's a MISS (The user (Player 1) has selected a different card (3 of Hearts) to that on CPU deck (2 of diamonds))

YOUR MISS POINTS ARE ZERO, IT'S GAME OVER

PRESS RESET TO TRY AGAIN