***Concentration* (*casino style*)**

Browser Game

1. Theme
    a. Casino
    b. 16 cards

2. User Stories
    a. As a player, I want to see a landing page with how to play instructions, so I get informed and prepare to play.
    b. As a player, I want to see clear labeled buttons, so I know what option I have to play.
    c. As a player, I want to click on any square on the board and see a card deck displayed, so I know my choice was registered.
    d. As a player, I want to be able to find matching pairs, and see the set display as I am search for another, so I visually track the matched ones.
    e. As a player, I want to see be presented with the final message when the timer is indicating the winner, so I know the outcome.
    f. As a player, I want to play another round so I have another go for fun.

3. Win/lose rules
    a. The player wins when all the pairs on the board are found withing 20 seconds.
    b. The player loses if the board is not completed Whitin 20 seconds
    c. The player loses after two wrong guesses.

PSEUDOCODE

----------CLI----------

//create repository, setup files: HTML, CSS, JS

/* ----------HTML----------*/

//boiler plate

//body

// div "instructions"

// h1 id="stateMessage"

// div*4 class="board"

// forEach div.board =div*4 id="square 0 -4"

//----------jS----------

1. constants

// define a constant variable for winning combos

//----------State Variables----------

// define variable for user's choice

// define variable for countdown

// define variable for message

// define variable for win

// define variable for lose

// define variable for mistakes

```
//-----------------Cache elements-------------------------
//Select the results displayed message
Select the countdown display
//----------------Event listeners----------------------------

// Delegated: add event listener to the parent element containing all the squares
// add event listener to startGame the timer button
// add event listener to reset button

//------------------Functions--------------------------------

// invoke the init function:
        // set all variables to initial state:
        // set countdown to 30 seconds;
        //set winner
        // using the event listeners setup, assign the player's choice to the player's choice variable
        // invoke get player function from game function

// invoke the primary render function
        // render the game message to the DOM

// after two player's clicks:
        // compare the first and second choices with the winning combos
        // IF the set matches any winning combo
        //THEN keep the choices visible
        //ELSE hide choices and add 1 to the mistakes variable and render a message
```

// check mistakes:

    // IF mistakes variable is less than 2 return/continue game

    //ELSE set looser to true and set message variable to "you reached two mistakes, you lost. Please try again"

// check board:

    // IF board is completed

        //THEN set win to true

// Render message to player.

    // run render function.

// when countdown reaches zero:

    // IF Board is incomplete set looser to true and set message variable to "you lost, try again"