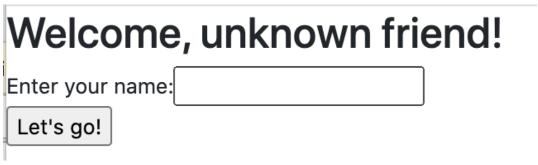
Implement a web version of the Memory game. In the game, the user has to discover pair of equal cards.

1) The webApp keeps a list of known users (not necessarily persistent: it may be in memory and last only as long as the webApp is alive). For every known user, the identity is kept in a cookie or session.
2) When a user accesses the webApp, two cases are possible:
   a. The user is known, so the webApp proceeds to point 3
   b. The user is not known to the system. In this case every request to the webApp redirects to a welcome page, which asks the username.



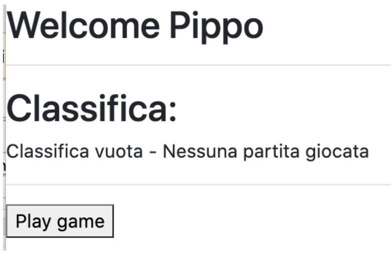
Welcome, unknown friend!

Enter your name:

If the name is not existing, it is added the users list, else the user is admitted with the declared username (we do not care too much about security here…).

3) Once the user is admitted, the initial page greets her/him, shows the ranking (Classifica) and a button “Play game” to start the game

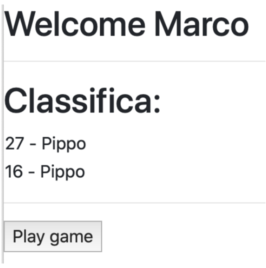
If no game has been played yet, the ranking says “Empty – no game played yet’, else the ranking shows the (up to) 5 best results, with score and name.



Welcome Pippo

Classifica:

Classifica vuota - Nessuna partita giocata



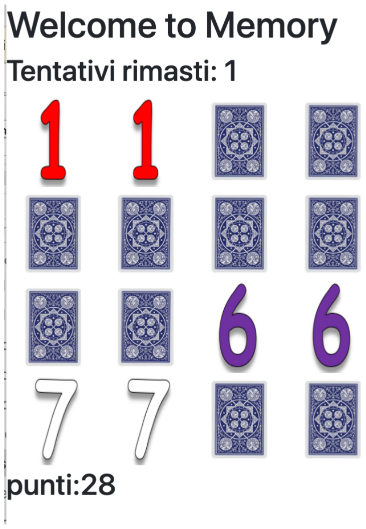
Welcome Marco

Classifica:

27 - Pippo

16 - Pippo

4) Pressing “Play game”, the game page is shown: it presents 16 covered cards in a 4x4 matrix. Each card has an unknown numeric value (1 to 8), so for each value two cards are present.



Welcome to Memory

Tentativi rimasti: 1

punti:28

5) The user has 4 attempts to find pairs. The number of available attempts is shown in the page (in figure, “Tentativi rimasti”).
6) When the user clicks on a card, it becomes uncovered and not clickable any more. The page is not reloaded, but only the card is uncovered (client side). The value of the card is not known client side, so the client interrogates the server by passing the card position (or identity) and getting back the card value.
7) When the user clicks on a second card, also this card is uncovered and becomes unclickable. The number of available attempts is decreased by 1. After that:
    a. If the two uncovered cards are equal, they remain uncovered and the user score (which is initially 0) in increased by a number of points equal to the value of the cards (if the two cards show 4, the user gets 2x4=8 points).
    b. If the two uncovered cards are different, the users score is decreased by 1, and after 1 second the two cards are covered, and become clickable again.
    c. In both case, the updated score is shown in the page (n figure above, “punti”).
8) When the number of available attempts becomes 0, all cards become unclickable, a label “Game over” appears and after 1 second the initial page (point 3) is shown with the updated ranking. NOTE: The ranking is obviously a general property of the webApp, keeping results of all users. Like the user list, it is not necessarily persistent.
9) To facilitate debugging and correction, a setting in the web-xml defines the mode, which can be TEST or PRODUCTION: in test mode cards are placed in a deterministic way (1,1,2,2,3,4…), in production mode they are placed randomly.

The game page has to be implemented in HTML+JS+XHR. The server side is minimal.

Useful hints:

- https://stackoverflow.com/questions/6764961/change-an-image-with-onclick/42226490
- https://stackoverflow.com/questions/14226803/wait-5-seconds-before-executing-next-line/14226826#14226826

The card images are available at https://latemar.science.unitn.it/esameWeb/img.zip