Distributed Programming I

Web Programming Test Assignment

Submission deadline: July 18, 2018, 11.59p.m. (on pad.polito.it and on cclix11.polito.it)

Build a website for playing the following game. The game manages a rectangular grid of cells whose dimensions are set by default to a width of 9 cells and a height of 7 cells. Each participant in the game can place rectangular shapes on the grid, which is shared by all players. The size of these shapes is fixed, by default, to 1x4 cells, and the shapes can be placed horizontally or vertically. The dimensions of the grid and the length (not the width) of the shapes must be configurable, simply by setting three variables in a single place of the PHP code. The shapes on the grid must be completely inside the grid and any two shapes on the grid must not be adjacent, i.e. there must always be empty cells around the shape, in each direction, including the diagonal.

The web application must have the following features:

1. On the home page of the site, anyone can view, without any registration, the grid. All the cells of the grid must appear with black-borders. Cells occupied by a shape must have a black filling, while free cells must have white filling. This view must not show the owner of each shape.

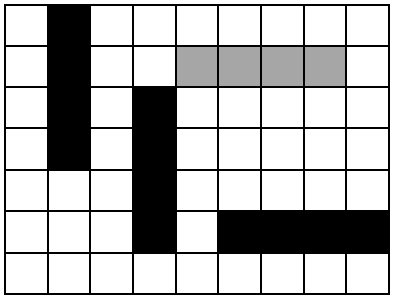
2. Each user can sign up freely on the site by providing a username, which must be a valid email address, and a password, which must contain at least 3 characters, including at least one non-alphanumeric character (i.e. neither a number nor a letter). In case of invalid username or password, the user must be notified by the client, before sending the data to the server, and signing up must be forbidden.

3. Each user can view, in his or her personal page, accessible only after authentication to him or her only, the grid with the shapes coloured according to their ownership: the shapes owned by the user must have green filling, those owned by the other users must have black filling, while free cells must have white filling. In his or her personal page, a player has the possibility to place new shapes, one by one. The placing of a new shape is done as follows: first the user has to click on two cells of the grid, so identifying the start cell and the end cell of the new shape to be placed. In this phase, the position of the new shape on the grid has to be shown by filling all the cells of the new shape with grey filling. In order to finalize the positioning of the new shape on the grid, the user has to click on a dedicated button, which causes the sending of the positioning data to the server. The system must notify the user by showing whether the operation has been successful, in which case the grid must be shown again with the grey-filled shape turned into a green-filled shape, or the operation has not been successful, in which case the grid must be shown again in its current status. Alternatively, if the user does not want to finalize the addition of the new shape, the user can click on another button, which has the effect of aborting the current operation and resetting the grey-filled cells to white filling. No shape can overlap with or touch any other shape on the grid, even owned by other users, according to the positioning rules previously expressed.

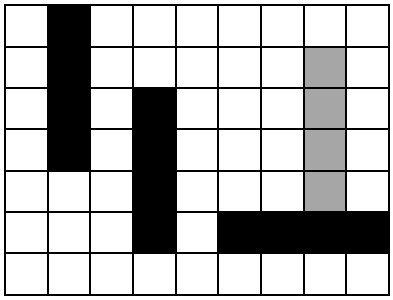
4. Each player can remove the shapes added by himself/herself, one by one, starting from the last added one, in reverse addition ordering, by clicking on another dedicated button one or more times. Each time the removal button is pressed, the cells occupied by the removed shape become free again, shown with white filling on the grid, and other players can potentially occupy them with their shapes.

5. Example:

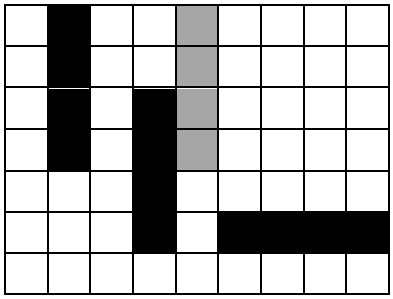
The following positioning on the grid is not possible because the new shape is adjacent to an already positioned shape diagonally.



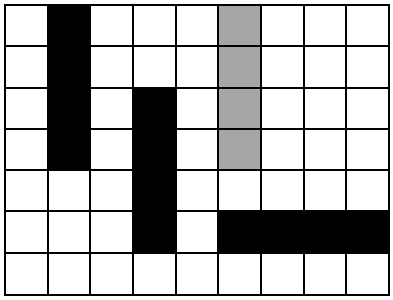
The following positioning on the grid is not possible because the short side of the new shape is adjacent to the long side of an already positioned shape.



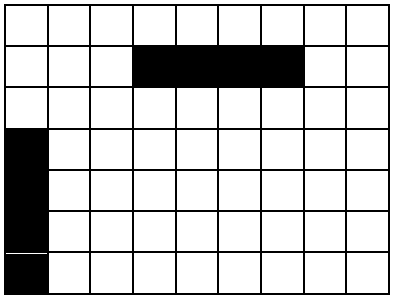
The following positioning on the grid is not possible because the new shape is partially adjacent to the long side of an already positioned shape.



The following positioning on the grid is possible, so when the user confirms it the cells can become green-filled.



6. In the application deployed onto the Labinf server there must already be 2 users with usernames *u1@p.it, u2@p.it*, and passwords p.1, p.2, respectively. Each user must have already placed a shape, as shown in the following grid status (the vertical shape has been placed by u1@p.it while the horizontal shape has been placed by u2@p.it).



7. Authentication through username and password remains valid if no more than two minutes have elapsed since the last page load. If a user tries to perform an operation that requires authentication after an idle time of more than 2 minutes, the operation has no effect and the user is forced to re-authenticate with username and password. The use of HTTPS must be enforced for sign up and authentication and in any part of the site that shows private information of an authenticated or signed up user.

8. The general layout of the web pages must contain: a header in the upper part, a navigation bar on the left side with links or buttons to carry out the possible operations and a central part which is used for the main operation.

9. Cookies and Javascript must be enabled, otherwise the website may not work properly (in that case, for what concerns cookies, the user must be alerted and the website navigation must be forbidden, for what concerns Javascript the user must be informed). Forms should be provided with small informational messages in order to explain the meaning of the different fields. These messages may be put within the fields themselves or may appear when the mouse pointer is over them.

10. The more uniform the views and the layouts are by varying the adopted browser, the better.

Submission instructions:
You need to submit your solution **both** on cclix11.polito.it **and** on pad.polito.it (as explained below).
The instructions already published in the Material folder of the course web page for the installation on cclix11.polito.it still hold. **Furthermore**, you need to submit your project (the same that you installed on cclix11) in a zip file named sXXXXXX.zip (without blank spaces in the name) to the following web site:
https://pad.polito.it/enginframe/dp1/dp1.xml (from inside the Politecnico network) or
https://pad.polito.it:8080/enginframe/dp1/dp1.xml (from outside).

In addition:

1. The sql script included in the zip file (submitted to pad.polito.it) to create the database must have a name with the following pattern: sXXXXXX.sql (where XXXXXX is your own student id).
2. The main page of your web site must be put in a file named index.html or index.php in your SECRET_FILDER such that the website can be accessed at the url http://cclix11.polito.it/~sXXXXXX/SECRET_FOLDER **without adding any other resource name at the end of the SECRET_FOLDER**.
3. DO NOT use absolute links

WARNING: The system that accepts your projects, works in an automatic way and it will stop accepting submissions at the scheduled deadline. For this reason, we recommend you DO NOT submit your work in the very last minutes before the final deadline.

In case of any doubt and question related to the project, please first visit the forum in the course website in order to check if other students have already asked the same question. Otherwise use the forum (not the teacher email) to ask your question so that the response will be available to all students.

The forum has to be used exclusively for requests of clarification about the text of the assignment and not for requesting help about how to solve it or how to solve specific problems encountered during the execution of the assignment.