Distributed Programming

Web Programming Test Assignment

Submission deadline 15th January 2016, h 23.59

The project consists in developing a simplified version of a web application for managing personal contacts, according to the following specifications.

1. All the users who access the website can see, without any authentication or registration mechanism, the number of persons signed up in the web site, and the number of contacts (but not the name) of the three persons who have the highest numbers of contacts, in decreasing order.

2. Each user can sign up freely on the web site by entering a username (in the form of an email address) and a password. The password must be repeated (i.e. written into two different fields). If the two copies of the password do not match the browser must show a warning to the user and the user must not be allowed to continue until the passwords match. If the username is already in use by another signed up user, the system must refuse signing up the user and must ask the user to choose another username, so that username uniqueness is always guaranteed.

3. An authenticated user must be able to see, in his/her personal page, his/her list of contacts and the number of contacts that each contact in his/her list has, including the user himself/herself. However, neither the name of these contacts nor any other identifier must be displayed. The number of each user's contacts must be stored efficiently, so as to avoid its recalculation at each display operation. An authenticated use must also be able to add a new person to his/her contacts. The new person can be specified by writing the username or by selecting it from a list that includes all the usernames of the persons who have the user who is trying to add them included in their contacts. The new person who has been added will find, in his/her contact list, the user who has added him/her. If the username that is added is not currently associated with a signed up user, a new user is signed up automatically with that username and the corresponding password is set to *pass*. The username is NOT case sensitive. Initially, the new user will have only one contact, that is, the one that just added this user. If instead a signed up user with that username already exists, no new user is signed up automatically, and the already signed up user is used.

4. An authenticated user u1 must be able to remove from his/her contacts any user u2 whom had been previously added by u1. Removing u2 from u1's contacts also implies removing u1 from u2's contacts.

5. In the submitted project there must be four signed up users with usernames alice@fb.it, bob@fb.it, carlo@fb.it and diana@fb.it and passwords p1, p2, p3, p4. Alice must have Bob, Carlo and Diana in her contacts, Bob must have Alice and Diana, Carlo must have Alice only, while Diana must have Alice and Bob.

6. Authentication by means of username and password must be done when necessary and must remain valid if no more than 2 minutes elapse between one page load and the next one. If a user requests one of the operations that require authentication after the deadline of 2 minutes since the previous page load, the operation must have no effect and the user must be forced to re-authenticate with username and password. Using HTTPS is mandatory during authentication and whenever the user is accessing a part of the application that is available only with authentication.

7. The general appearance of the web pages must include: a header in the upper part of the page, a navigation bar on the left side with all the links needed to perform the various operations, and a central part which is used for the main operation.

8. 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.

9. The graphical layout, even if simple, must be consistent, that is, the pages must be as much as possible uniform among all the different browsers.

10. Extra requirement (only for students having the 8-credits course): Design an XML format that can be used for storing a list of contacts, and specify the format using a schema. Store the schema in a file named `schema.xsd`, and store the file in the main folder of your application (the same where you have stored your `index.html` or your `index.php` page), so that it can be freely downloaded. Introduce a mechanism that periodically updates the list of contacts displayed by a user, even if the user is not performing operations, without reloading the entire page. This can be achieved in a simple way by issuing periodic AJAX requests (for example, a request every 30 seconds).

Submission instructions:
The instructions already published in the Material folder of the course web page for the installation on the 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 firstly 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.