Distributed Programming I

Web Programming Test Assignment

Submission deadline June 20, 2016, 11.59 p.m.

The candidate is requested to build a simplified version of a web application for managing the reservations of a set of machines (for example 3D printers). The web site must follow the following requirements:

1. Every user can register freely on the web site (providing name, last name and a valid email - the latter to be used as username). In any page of the site, even without authentication, anyone shall be able to see the current state of all reservations (starting time, duration time and selected machine) of all users for all machines. It is necessary that in such view, the reservations will appear ordered by reservation's staring time.

2. A user, once authenticated, can add a new reservation or remove an existing one; to this aim, a list of the user's reservations shall be displayed, in order to let the user select the one he/she wants to remove. With each reservation, a duration is associated (an integer number expressing the minutes) and a starting time (represented by the pair hours:minutes, with hours varying between 0 and 23 and minutes varying between 0 and 59). For the sake of simplicity, days are not considered here, that is, assume that all reservations are for the same day, not further specified. The list of reservations presented to the user must be ordered by reservation's starting time. The user shall be able to remove her reservation from the site, but not before that 1 minute has elapsed since the reservation start time.

3. The site must consent to reserve a machine to a user as long as there are available machines to perform the task. Since the machines are all equal and they perform the same kind of job, it is possible to have multiple reservations that overlap in time, as long as the number of machines is sufficient. The criterion for assigning a machine to a job can be decided freely and the assignment must be done automatically by the site. The number of available machines is fixed to 4, however, it must be possible to change this number by modifying a PHP variable in a single point of the source code of the web site. If all the machines are busy for a part or the whole period requested by a new reservation, this reservation must be rejected.

4. In the project to be submitted, three users must already exist: *u1@p.it*, *u2@p.it, u3@p.it* with passwords *p, p2 and p3* respectively. Each one of them must have already made two reservations for two distinct machines and in a period of time that does not overlap with any of the other reservations. Moreover, both *u1* and *u2* must have another reservation, not overlapped with any of the above mentioned reservations, but these reservations must be partially overlapped between each other (Thus, the total number of reservations that must be already present is 8).

5. Authentication by means of username and password must be done when requested by the user 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 has no effect and the user is 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.

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

7. 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.

8. 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.

9. Extra requirement (only for students having the 8-credits course):

The client side of the application must use AJAX in order to constantly keep the list of reservations up to date: every 5 seconds a new AJAX request must be sent to the server in order to check if the list of reservations is up to date and, if it is not, to download the last updates. In order to maximize efficiency, the downloaded updates must include only the operations to be performed on the list (e.g. delete a particular list item or add a particular list item) rather than the new full list, and the web page must be updated accordingly, without re-loading the whole page. In order to make this possible, the server can keep a (numbered) list of the last operations performed and the number of the last operation already sent to the client can be kept in the client session or communicated by the client. For efficiency reasons, the server has to delete old operations. The information about updates must be encoded in an XML format. For this purpose, a schema must be designed. This schema must be available for download from the web site with name schema.xsd (in the same location where the main index.html or index.php is placed).

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.

The forum is 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.