

Distributed Programming I
Web Programming Test Assignment

Submission deadlines: September 1, 2017, 11.59p.m. (on pad.polito.it)

AND September 4, 2017, 11.59p.m. (on cclix11.polito.it)

Build a simplified version of a website for managing the preferences of students for booking a seat in two out of the three exam calls (#1, #2, #3) of a course in a University. Each exam call has, initially, 3 available seats, and each student can express 2 preferences when trying to book seats. For the sake of simplicity, the web site considers only one course.

The website must have the following features:

1. On the home page of the site, one can view, without any registration, the list of students booked for each call. For each booking, the email of the student who made it is shown, and the bookings are shown in the same order they were made. In addition, the home page of the site must show the total number of bookings made so far for each call and the grand total of bookings for the course.
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 be requested 2 times during the sign-up procedure. While the first password is entered, a coloured text must be shown which notifies the strength of the password, determined in the following way: if the password length is less than or equal to 2 characters: “weak”, written in red; if instead the password length is greater than 2 characters, if it contains only letters, without digits or other characters such as punctuation characters: “medium”, written in yellow; otherwise “robust”, written in green. The notification text must be updated each time a character is entered or eliminated in the password text field, without server intervention. All passwords must be accepted (even weak and medium ones).
3. Each user can request, in his or her personal page, accessible only after authentication, two exam calls (for example, first preference: #3, second preference: #1). Once both preferences have been entered (it is compulsory to enter both in order to proceed), they are sent to the server. The send operation is triggered by a SINGLE action of the user (e.g. pressure of a button), and the server will communicate, in the response, what has been actually booked. The result depends on the application of the booking algorithm described in the following point. The booking operation can be cancelled, but only in its entirety (i.e. not for a single exam call). When there are no bookings for a student, the student can set two new preferences and request the booking.
4. The requested seats are assigned to a student according to the following algorithm: if there are seats available in the selected exam call, a seat is assigned to the requester. The booking operations for the first and second preferences are performed both and sequentially in the server, starting with the first preference, but letting other users who are booking simultaneously to insert their booking attempt before the booking attempt for the second preference. If there are no available seats in both selected exam calls, the student is assigned no seat, and this condition is notified to the user in his/her personal page by means of a message. Similarly, if seats are assigned to the student, they are communicated to the student.
5. The bookings must be stored in the DB in order to avoid to recalculate them each time.
6. Example:

Let us assume there are 6 users A, B, C, D, E, F. Initially no one of them has made bookings.

A requests #3 and #1. Both seats are assigned. State: #1: A; #2: -; #3: A

B requests #3 and #2. Both seats are assigned. State: #1: A; #2: B; #3: A,B

C requests #1 and #3. Both seats are assigned. State: #1: A,C; #2: B; #3: A,B,C

D requests #3 and #1. Only #1 is assigned. State: #1: A,C,D; #2: B; #3: A,B,C

B cancels the booking made. Both booked seats are removed. State: #1: A,C,D; #2: -; #3: A,C

E requests #2 and #3. Both seats are assigned. State: #1: A,C,D; #2: E; #3: A,C,E

D cancels the booking made. The only booked seat is removed. State: #1: A,C; #2: E; #3: A,C,E

B requests #3 and #2. Only #2 is assigned. State: #1: A,C; #2: E,B; #3: A,C,E

D requests #2 and #1. Both seats are assigned. State: #1: A,C,D; #2: E,B,D; #3: A,C,E

Note that if F had attempted to book, simultaneously with D, #1 and #2, then the final state could have been: #1: A,C,F; #2: E,B,D; #3: A,C,E where both D and F would have their first preference satisfied.

7. In the submitted project, 6 users A,B,C,D,E,F with usernames *a@p.it*, *b@p.it*, *c@p.it*, *d@p.it*, *e@p.it*, *f@p.it*, and, respectively, passwords *pa1*, *pa2*, *pa3*, *pa4*, *pa5*, *pa6*, must be present. These users must have made booking as in the example, reaching the state: #1: A,C,D; #2: E,B,D; #3: A,C,E (then, F exists as a user but has made no bookings so far).

8. Authentication through username and password remains valid if no more than two minutes have elapsed since the last page load. If a user attempts 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 refers to an operation of an authenticated user.

9. 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.

10. 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; if Javascript and cookies are both disabled, the user needs not be alerted). 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.

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

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/engineframe/dp1/dp1.xml> (from inside the Politecnico network) or
<https://pad.polito.it:8080/engineframe/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_FOLDER 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 order to be admitted the network programming test on September 5 2017 it is **REQUIRED** to submit the solution of this assignment on pad.polito.it by the deadline of September 1 2017. This solution can be slightly changed when loaded onto cclix11.polito.it in order to fix small errors or malfunctionings.

We remind you that cclix11.polito.it is accessible not only from LABINF, which will remain closed till September 3 2017, but also from the WiFi of the Politecnico campus (eduroam network).

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 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