Distributed Programming I

Web Programming Test Assignment

Submission deadline: July 7th, 2019, 11.59p.m. (on pad.polito.it and on cclix11.polito.it)

Build a simplified version of a website for renting personal transportation means (i.e., bicycles, motorbikes) in specific renting locations in a city. Assume that each city has a rectangular shape and the renting locations are identified as specific points (X,Y coordinates, which are integer numbers) inside the rectangular map. Each customer uses the booking system available in the website, which must have the following features:

1. On the home page of the site, anyone can view, without any registration, the map of the city with all the renting locations, depicted as small circles in the map, each one with a diameter of 5 pixels. The filling colour of the circle is green if the renting location has at least 4 available means, yellow if it has between 1 and 3 available means, red otherwise. The home page of the website shows also the overall number of means available in the city, separated for bicycles and motorbikes. By clicking on each renting location on the map, the location is selected (an additional black circle appears to highlight the selected point) and, without reloading the page, the application shows, on one side of the map, the total number of bicycles and motorbikes available in that location. By clicking again on the point, the renting location will be unchecked, and the relative information on the side of the map will disappear. If another point is selected on the map, the previous selection is automatically removed before activating the new selection.

2. Each customer who is already registered in the system can log into the application by means of a dedicated form. A new client willing to use the site 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 2 special characters (i.e. non-alphanumeric characters). The password entered by the user must be confirmed by the user by filling a second identical password field in the form. If the two passwords entered by the user do not match, sign-up must not continue. If sign-up is successful, the new client must be automatically logged into the system without having to perform a regular sign-in nor any other action.

3. Once logged in, a customer must see the overall map including all the renting locations, as in the home page of the website, with the possibility of showing the availability of each location by selecting it on the map.

4. Once a location has been selected, a logged-in user can choose, using a specific form, how many personal transportation means (of each type) he wants to rent according to the following rules. While choosing the vehicles, no information must be sent to the server. Only at the end of the choice, the entire request has to be sent to the server that will process it according to the current availability for that location, which in the meantime could have been changed. The server has to process the request according to the following logic: if the requested vehicles are still available, they are reserved for that user. If there are not enough motorbikes, the server will reserve a number of additional bicycles in order to keep the total number of booked vehicles unchanged. If there are not enough bicycles, they CANNOT be compensated with motorbikes. If there are not enough vehicles, possibly converted with the previous logic, the booking must not take place. In all cases the server will show the user a message that details what happened.

5. Example (to be used in the submitted project)

Three customers are already signed-up, with usernames u1@p.it, u2@p.it, u3@p.it and passwords p.-1, p.-2, p.-3 respectively, who have already performed the operations described below. Use a map of 600x400 pixels, with 6 different renting locations, with coordinates X, Y (Y grows from the bottom to the top) equal to:

10,20: 2 bicycles and 3 motorbikes (green circle)

30,60: 3 bicycles and 4 motorbikes (green circle)

40,134: 3 bicycles and 3 motorbikes (green circle)

30,180: 2 bicycles and 1 motorbikes (yellow circle)

130,290: 3 bicycles and 0 motorbikes (yellow circle)

255,360: 4 bicycles and 2 motorbikes (green circle)

440,241: 0 bicycles and 0 motorbikes (red circle)

User u1@p.it books 4 motorbikes and 1 bicycle for the 40,134 point: the server proceeds with the booking assigning 3 motorcycles and 2 bicycles. The renting location remains with 1 bicycle and zero motorbikes, it is then shown in yellow.

User u2@p.it books 3 bikes for the 30,180 point: the server indicates that it is not possible to proceed with the booking.

6. In the submitted project there must already be three customers present, u1@p.it, u2@p.it, u3@p.it with password p.-1, p.-2, p.-3, which are in the situation at the end of the previous example.

7. Authentication or registration through username and password remains valid until the user has periods of inactivity exceeding 2 minutes. If a user attempts to perform an operation that requires registration or authentication after the inactivity has exceeded 2 minutes, the operation has no effect and the user is forced to re-authenticate with a username and password. The use of the HTTPS protocol for registration and authentication must be imposed in every part of the site that shows information relating to a registered or authenticated user.

8. The general appearance of the website must contain: a header at the top, a navigation bar on the left side with links or buttons to be able to perform the various possible operations, and a central part where the main operation takes place.

9. Cookies and JavaScript must be enabled, otherwise the site should not work (the user must be notified on the main page, and the webpages must not be displayed). Forms must include short explanatory messages on the meaning of the various fields, which may be present within the fields themselves or appear when the pointer is over the field itself.

10. You must ensure that the visual appearance of the website, even if simple, will be as uniform as possible even if the browser used changes.

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.