Web Development
Assignment 2

Semester 1, 2017

DUE ON: 11:59 PM, Thursday 01 June 2017
Assignment Worth: 25% of total marks

NB: Assignments will be accepted up to five (5) days late, but a penalty of 5% per day (or part of a day) late will be imposed on either the team or the individual depending on the circumstances.

NB: Students are referred to the school's policy on plagiarism. A confirmed case will incur zero mark to all the involved students.

1. Assignment requirement specifications:

This is an individual assignment. Students are referred to the school's policy on plagiarism.

The aim of this assignment is to develop a better understanding of building web applications using simple Ajax techniques, and PHP and MySQL on the server. It assumes that you are able to work with MySQL and PHP on the server, as learned in the first part of this course, and that you can develop an Ajax client interface using HTML and JavaScript, as learned in the second part of this course.

For this assignment, you will need to create all the appropriate HTML, PHP files and Database tables. You should save and test all your HTML and PHP files on *cmslamp14.aut.ac.nz* server in a directory " */home/<your student ID>/public_html/assign2*".

2. Assignment Tasks

The assignment is to develop simple web-based taxi booking system called CabsOnline. CabsOnline allows passengers to book taxi services from any of their internet connected computers or mobile phones. The techniques you are going to use include the Ajax techniques (JavaScript/HTML, XMLHttpRequest, CSS, and DOM), MySQL and PHP. For client-server communication, you must use XMLHttpRequest object.

Two components (booking and admin) of such an online service that must be completed for this assignment are specified in the following two sub-sections. Other components such as querying service for drivers, monitoring services for customers, payment processing, detailed processing for assigning taxi are not required in this assignment but you are free to extend for your fun later.

2.1 Booking page (booking.htm)

This component is used to allow a passenger to put in a taxi booking request in Auckland and surrounding areas. The inputs for such a request include customer name, contact phone, pick-up address (unit number if applicable, street number, street name, and suburb), destination suburb, and pick-up date/time. Some other details such as number of passengers, car type, building type etc. may not be required for this assignment. Once you get these inputs, you need to generate a unique booking reference number, booking date/time (i.e., the date/time the booking is made) and status with initial value "unassigned" for the request, add the request in a MySQL database at the server side. The specific functions of this component include

1) Design and create a MySQL table for storing information of all requests. For each request, you need to store the generated booking number, customer's inputs, the generated booking date/time, and the generated status.
2) On the client side, design an HTML page (booking.htm) and a client-side program to take inputs for a booking request, and pass them to server for processing (using XHR object).

You need to validate these inputs before sending them to the server. All input items except unit number must be provided. The pick-up date/time must be no earlier than current date/time.

3) On the server side, for each booking request, a server-side program is required to generate a unique booking reference number, booking date/time and a status with initial value "unassigned", and add them together with customer's inputs into the MySQL table. In addition, it is also required to return confirmation information to the client.
4) On the client side, upon receiving the returned information from the server, the client-side program needs to display the returned confirmation information "Thank you! Your booking reference number is <bookingRefNo>. You will be picked up in front of your provided address at <pickupTime> on <pickupDate>." in a suitable area of the HTML page booking.htm.

2.2 Admin page (admin.htm)

This component allows administrative people of CabsOnline to view those taxi booking requests that need to be assigned as soon as possible and to assign taxi for a particular booking request. Note authentication is not required though it is necessary in the real application. If you provide this function, you must provide the password and explain it in the readme document. The specific functions of this component include

5) Design an HTML page (admin.htm) that takes two types of requests. The first is to search some booking requests and is realized by a single button to show pick-up requests. The second is to assign taxi for a particular booking request and is implemented by a text box allowing input for a specific booking reference number and followed by a button assign taxi. Once the button show pick-up requests is pressed, a client-side program issues a query request to the server; once the button assign taxi is pressed, a client program issues an assigning request to the server with an inputted booking reference number.
6) On the server side, a server-side program responds differently for different types of requests from the client. (a) For a request for showing pick-up requests, the server-side program executes a query on the MySQL database for finding those "unsigned" booking requests with a pick-up time within 2 hours from now only. For each found request, the booking reference number, customer name, contact phone, pick-up suburb, destination suburb, and pick-up date/time are required to return to the client. (b) For a request for assigning taxi, make an update to the MySQL database to change the status of the booking request that matches the given booking reference number from "unassigned" to "assigned", and return confirmation information to the client.
7) On the client side, for a request for showing pick-up requests, you display the returned information in a suitable area of the HTML page admin.htm. (a) For a request for showing pick-up requests, you display each found request in a separate line. It is okay to display each line in plain text. However, you are encouraged to display all requests in a table. (b) For a request for assigning taxi, you display the confirmation information "The booking request <bookingRefNumber> has been properly assigned".

Submission Requirements

You should ensure that all files used for the assignment sit in a directory called "*assign2*" (use this name exactly, it is case sensitive and no space between *assign* and *2*) within your *cmslamp14* account. The directory should contain no other files and no other sub-directories (i.e., all files are placed directly under the "*assign2*" directory).

The assignment should be submitted as an individual work using the Assignment 2 link in AUTonline (inside the Assignments area)

Please compress all your html and PHP files as a zip file named "assignment_2.zip" to

submit via the AUTonline submission link. You can submit more than once.
The files should include:

- two HTML files booking.htm and admin.htm;
- any JavaScript files that you use;
- any PHP files that you use;
- a text file that includes the MySQL commands that you used to create the database;
- a file readme.doc that includes
  - a list of all the files in the system;
  - brief instructions on how to use the system.

For each submitted file, we require clear comments including student information, description of the file, and description of each function defined in this file.

The MySQL database that you use should be constructed in your *cmslamp14* account. After submission, you are not allowed to change any of the submitted files in the *assign2* directory on your *cmslamp14* account; time stamps will be checked.

Assignments that fail to follow "submission requirements" will NOT be assessed.

Marking Scheme

Work will be assessed based on the quality and presentation. The assignment will be marked out of 40 and will contribute 25% towards assessment of this course.

Assessment item	Marks
comment and readme.doc	4
Quality of code	3
Usability of user interface	2
2.1.1	3
2.1.2	4
2.1.3	6
2.1.4	4
2.2.1	4
2.2.2	6
2.2.3	4
Total	40

** If your assignment cannot run on cmslamp14, your result will be 0 marks for this assignment.*