

# Assignment 2: Implementing a dynamic web site

Due: 9:00am **Friday 21<sup>st</sup>** Monday 24<sup>th</sup> September 2018

## Introduction

You are to apply all the Web technologies (DOM, XML, Ajax, PHP, Oracle, etc.) learnt from lectures and practicals to make the site that you created in [Assignment One](#) *fully functional*. That is,

- To validate the **checkout form** using Ajax and PHP **on server side**. User's input is validated on the fly. If error is found, appropriate information is displayed.
  - All required fields should be checked for completion, and all data that are entered should be checked for 'correctness'. For example, buyer contact details (name, phone number, email address, etc.), delivery address (unit number/street name, city, postal code, etc.), delivery method selected and payment methods (credit card number, expiry date, card holder name, CVV, etc.) are fields expected in the checkout form to be validated.
  - Regular expression is recommended (NOT compulsory) for pattern matching in PHP.
  - You decide what validation is required and how the validation is performed. For instance,
    - All required fields checked for completion
    - the fields checked for a specific length, numeric data or a predetermined format
    - the messages provided help the user fix the 'problem'
- To develop the system to respond to the forms by accessing your products/services XML files, querying your **Oracle** database, generating contents, and adding information to the database. Your PHP code needs to be used to enable
  - search suggestions: a suggest/autocomplete box similar to Google Suggest, where a user gets suggested products/services while typing in the search form.

Example: User is typing a brand/product name:

Emma

Suggestions which would be displayed:

Emma Tub Armchair

- search function: the **search form** responds to user's input with retrieved information from server side (using xml or database). When user clicks on "search" button, a search result page is loaded. If a product link is clicked from suggestion list or result page, user is led to the product detail page, where user can add the product to a shopping cart.
  - shopping cart: it allows a user to place items in the cart and calculates a total for the order, including shipping and handling (i.e. postage and packing) charges and the associated taxes (e.g. GST), as applicable upon checkout. After checkout, all the data from the completed form will be inserted into the relevant table. A receipt page confirming the order that has been made should be displayed or in the event of an error, an appropriate error message should be shown to the user.
- To implement the "login" and "register" function on your website.

- When a new user makes an order, a new account/username and a temporary password will be created automatically after order confirmation. User's personal details will be saved in database under the created account/username.
- Once logged on, on "my account" page, a user can change password and edit personal details.
- Once logged on, user's registered name should be displayed at a proper location on the pages accordingly.
- When a logged on user makes an order, the shipping details in the checkout form can be auto-filled directly with the information collected on "my account" page. The details should be editable.

**You are permitted to change the pages provided by the templates if you believe that they can be improved for the purposes of this assignment. However before receiving your assignment 1 mark, please do not modify the original html pages. Copy them to another folder and then copy them back later after the assignment 1 mark be finalized.**

## Requirements

General requirements include:

- *Well developed and structured PHP code.* Write functions to re-use code for common tasks. The code should be clear and properly indented; use meaningful variable names; and be free of errors.
- *Well documented code.* Appropriate and consistent comments should be placed throughout the code. The comments should be sufficiently detailed to allow an assessor to understand what it is doing without having to spend too much time on your code.

**Assignment Project Exam Help**

**<https://powcoder.com>**

**Add WeChat powcoder**

## Suggested progress

The workload of this assignment is extremely **heavy**. You are highly recommended to start it as early as possible. It is suggested that:

- By the end of week 7, you'd better have completed checkout form validation.
- By the end of week 8, you'd better have completed search suggestion, search function and shopping cart.
- By the end of week 9, you'd better have completed register and login relevant functions.
- In week 10, wrap up.

## Submission

The Assignment Two is worth **50%** of the overall assessment for this unit. Your submissions **MUST** use the Dropbox that will be placed on CloudDeakin. Please follow the instructions to submit your assignment on clouddeakin. Your website created for SIT203 assignments must be hosted by Deakin Web server (<http://www.deakin.edu.au/>). **A website hosted by a 3rd party server will not be accepted.**

**Email submission will not be accepted.**

Your assignment will be marked after the due date and the assignment result (with comments) will be sent back to you.

## **Extension**

If you wish to apply for an extension, please contact your campus lecturer AT LEAST TWO DAYS before the due date; also attach what you have done. LAST MINUTE EXTENSION REQUEST WILL NOT BE GRANTED. The teaching staff will then make a judgement as to whether an extension shall be granted.

---

**Assignment Project Exam Help**

**<https://powcoder.com>**

**Add WeChat powcoder**