Department of Information and Technology
(Lee Wai Lee)

Higher Diploma in Software Engineering (IT114105)

Module Name   : Internet & Multimedia Applications Development
Module Code   : ITP4513
Submission Deadline : Phase 1: **Week 42, before 11:55 pm, 21 Jun 2019, Friday**
                           Phase 2: **Week 45, before 11:55 pm, 8 Jul 2019, Monday**
Hand in Methods   : *Refer to the instructions at Moodle*
This Group Project : 30% of total module marks (*also it is part of EA components*)

> **The result of EA will not be counted if you do not meet the minimum 70% attendance requirement (if any) governed by the general academic regulations of your programme/course unless approval of the campus principal has been granted.**

1. Objectives
In this project, student are asked to:
- build a web application which provide different functions for the dealer and administrator.
- apply software development skills to develop a website which is user-friendly, interactive, robust and easy to maintain.
- apply the knowledge that you learned in this module to solve the tasks. (e.g. HTML, CSS, JavaScript, PHP and simple SQL commands)

2. A simplified procedure to show how the web application will be used
There are two user roles for the Spares Order System:
a. Dealer can make orders and manage their orders.
b. Administrator can manage the parts and the orders of dealer.

3. Design for Dealer (Interface Design: 15 marks; Function: 35 marks)

a. Register account into system
Managed to create a new dealer account into the system with all necessary information.

Required information during registration as below:
1. Dealer's ID
2. Dealer's Name
3. Phone Number
4. Address
5. Password

* dealerID should be created by dealer as Primary Key

b. Update dealer's profile
Managed to view and update dealer's information

Dealer's information including:
1. Name
2. Address
3. Phone Number
4. Password

*dealerID cannot be changed by the dealer

c. Make the orders
Managed to make orders into the system with all necessary information

View product information
- List of parts shown on site for dealer to choose (show the parts when status is "Available" and the stock quantity should be greater than zero)

The parts should include 2 stock status:
1. "Available" : Show only the available parts.
2. "Unavailable": The parts are unavailable.

Required information for creating an order
1. Delivery Address:
   - The address default as it registered.
2. Part Name & Quantity:
   - Being able to order more than one parts in the same order.
   - Order quantity cannot be greater than the stock quantity in Part table

*orderID should be generate automatically by the system as Primary Key

d. <u>View & update order records</u>
   Managed to view the orders with necessary information

   Required information for viewing the orders
   1. Order ID
   2. Address
      - Dealer address (Default) /New delivery address
   3. Order Date
   4. Part Number
   5. Part name
   6. Quantity
   7. Status
   8. Total Price
      ,etc.

   The site should include 4 statuses:
   1. "In processing": The order was made
   2. "Delivery": The parts are delivering to the dealer
   3. "Completed": The dealer received the parts ordered
   4. "Canceled": The order is canceled by administrator

   Required functions:
   1. Being able to cancel the order even if the status is "In processing" stage
      - Records should be able to delete from the database
   2. Being able to make Confirmation when all parts are received by dealer
      - e.g. Click buttons to confirm completed orders
      - Order status should be able to change from "Delivery" to "Completed"

4. Design for Administrator (Interface Design: 15 marks; Function: 35 marks)

a. Login to system
   Managed to login into the system with all necessary information.

   Required information during login
   1. Email
   2. Password

b. View and edit parts' information
   Managed to create a page contain the list of parts

   Required part information:
   1. Part Number
   2. Part Name
   3. Stock Quantity
   4. Stock Price
   5. Display the email of the administrator who did the latest editing
      ,etc.

   *partNumber should be generate automatically by the system as Primary Key.

   Required functions:
   - Create an "edit" button to make change of stock quantity and stock price
   - Administrator can add in new parts /remove the parts (change
     the stock status from "Available" to "Unavailable")
   - The latest modified email will be stored in Part table when the related administrator
     add/update the part record

c. Manage order
   Managed to view the order with necessary information.

   Required information for order page
   1. Order ID
   2. Dealer ID
   3. Dealer Name
   4. Order Date
   5. Address
   6. Part Number
   7. Quantity
   8. Total Price
      ,etc.

   Required Functions:
   - Deliver the parts when they are ready to deliver.
     → Change the order status from "In Processing" to "Delivery"
   - → Deduct the quantity of delivered part from Part table
   - Only dealer can change the "Delivery" stage to "Completed" stage, after they receive the
     goods

   Required functions:
   - Being able to cancel order if the dealer wish to
     → Change the order status to "Canceled"

5. **Form your project group**

    Each student needs to form a project group, **the maximum number of students in each group is 2.**
    I strongly recommend you to form a group to complete this project as you can benefit from sharing
    skills/codes amongst your members, and you can learn to plan, coordinate, and integrate work done
    by each member.

    Study carefully the given ERD and table structures before you start the implementation.

6. **Additional requirements of your project**

a. Your web site should only use PHP as the server-side programming language (i.e. not ASP, ASP.NET,
    JSP, servlet etc.), however, you may use JavaScript and CSS for specific purposes. The database server
    used must be mySQL (version 5.0 or above).

b. In your PHP code, you must ensure to use the following *parameter values* for the following mySQL
    database functions :
    $conn = *mysqli_connect*($hostname, $username, $password, $database);
    set to the values below in a PHP script which is *shared by* the web pages :
    $hostname = "**127.0.0.1**";
    $database = "**projectDB**";
    $username = "**root**";
    $password = "";

7. **Items to submit (Phase 1)(30% of total project marks)**
    Submit all UI design using CSS and HTML. Submit to Moodle on/before 11:55 pm, 21 Jun 2019,
    Friday.

8. **Items to submit (Phase 2)(70% of total project marks)**
a. A ***CD-ROM*** or ***DVD-ROM*** or ***USB*** which stores a ***softcopy of all files*** for the whole web site. All files
    must be stored in **non-compressed format** (no .zip or .rar files please !)
b. provide a SQL script file ***CreateProjectDB.sql*** to let the lecturer to re-create the database and test
    data
c. for the SQL script file ***CreateProjectDB.sql***, it must contain *CREATE TABLE* commands to setup the
    database tables in **projectDB** database. Include necessary *INSERT* statements to add additional
    sample records you want to provide. The following is a sample SQL script :

```
drop database IF EXISTS projectDB;
create database projectDB character set utf8;
use projectDB;

drop table IF EXISTS Users;

Create table Users (
  userName Varchar(30) NOT NULL,
  userPswd Varchar(10),
 Primary Key (userName)) ENGINE = InnoDB;

INSERT INTO Users (userName, userPswd) VALUES
('admin1', 'secret1'),
('admin2', 'secret2');
```

> You must specify the **InnoDB** engine for a database table :
> ENGINE = **InnoDB**
> Full explanation of different ***mySQL database engines*** :
> http://dev.mysql.com/doc/refman/5.0/en/storage-engines.html

d. a ***demonstration*** of your completed web site should be recorded by
    a *30-day free-trial software Camtasia Studi*o 8
    (http://discover.techsmith.com/try-camtasia/clkn/https/www.techsmith.com/download/camtasia/).
    You should save different parts of your demonstration into different **.mp4** files. In a *Word* document
    named **video_list.docx**, briefly describe the main content of each demo video file you have created.
    The video files will facilitate the lecturer to have in-depth evaluation of your web application. Here
    are some online tutorials for **Camtasia Studio 8** http://www.techsmith.com/tutorial-camtasia-
    current.html :
    Getting Started: 1 - Record Full Screen :
    http://www.techsmith.com/tutorial-camtasia-record-full-screen.html
    Produce and Share an MP4 Video :
    http://www.techsmith.com/tutorial-camtasia-produce-and-share-mp4-video.html

9. **Assessment criteria of your project**

a. The functions implemented can perform correctly in *general* and *special* situations
b. *Enough detail* of database records and extensive *data validation*
c. Techniques used to promote *code reusability* (e.g. share common PHP/JavaScript/CSS files amongst different web pages) and *standardize the user-interface* of the web pages
d. Coding style (e.g. indentation, meaningful variable names, modularity by user-defined functions etc.) and meaningful *comment* is added to program codes
e. *Creativity* to enhance implemented functions so that they become easy to use, more interactive to the users or can handle some problems in real life situation
f. Screen design and overall *quality of the integration* of different functions in the web site

10. A guideline for web development
It is a step-by-step approach I suggested for inexperienced web developers to develop the web site easily :
- decide what information to be displayed and design a number of web pages in HTML code (not PHP code at this stage) to display the information
- think about the site structure by creating different sub-folders to store files of different purposes (e.g. **images** folder to store image files, **style** folder to store CSS files, **Connections** folder to store files which define the settings for database connection) and design the linkages between the pages. You can easily view the site structure using DW8's site map view
- create HTML web pages (don't add JavaScript so soon) and design the layout with HTML codes and CSS rules. It is a good practice to check your .html files can pass the XHTML validation after you complete a .html file
- when using CSS, it is preferred to create *external CSS files* (stylesheets) which can be reused in other web pages, so that other pages can have consistent formatting
- use DW CS6's template features which can help you to create a new page with a standard layout and also it provides common editable regions for web pages created from the same template.
- define frameset(s) and navigation bar or menu to link up different pages
- add JavaScript code to produce more interactive behaviors (such as validate data in the form, highlight a table row with different background color when the mouse move over a table row). It is preferred to use *external JavaScript file* which will be reused in other web pages
- replace hyperlink text with image / button to beautify the links. Dreamweaver can help you to create nice Flash buttons easily
- finally, it comes to the hardest work, that is to convert some of the HTML codes into PHP codes in order to generate dynamic contents from data extracted from database, cookie and PHP pre-defined arrays ($_POST, $_GET, $_COOKIE, $_SESSION, $_FILES, $_SERVER etc.)

11. Penalty for plagiarism
- Each student has to submit his/her own work. Plagiarism (抄襲) will be treated seriously.
- All group projects that have been found involved wholly or partly in plagiarism (no matter these projects are from the original authors or from the plagiarists) will score ZERO marks. Furthermore, disciplinary action will be followed.

Late submission will receive ZERO marks