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	:	Week 47, **before 11:55pm, 18 Jul 2017, Tuesday**
Hand in Methods	:	*To be announced by the lecturer*
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 you are asked to :

- build a web application which provides different functions for *runners*, *volunteer, administrators* and *sponsors*
- apply software development skills to develop a web site which is user-friendly, interactive, robust and easy to maintain
- apply the knowledge you learned in this module to solve tasks which require your skills in HTML, CSS, JavaScript, PHP and simple SQL commands

2. A simplified scenario to show how the web application will be used

a. There are four user roles for this Online Marathon Skills management system : *runners*, *volunteer, administrators* and *sponsors*.
b. Runners can register to marathon events and view their records.
c. Volunteers can manage runner event records and Race Kit records.
d. Administrators can create new events and manage volunteer, runner and charity records.
e. Sponsors can view the runner event record and add sponsorship to new events.

In the following table, the functions designed for each user role are only accessible by a logged in user of the correct user role, unless it is specified in the described function.

3. Functions for the Runners *(25 marks)*

Done By

a. Register to System

Can create a new Runner account into the system with all necessary information.

Required information during registration

Password, First Name, Last Name, Gender, Date of Birth, Email, Country

Optional information & can be updated after registration

Profile Picture (a file path to the uploaded picture)

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

*RunnerID will be shown when registration completed for login purpose.

b. Update Runner Profile & Check Event Record

Can view and update his/her personal information and view event records.

Personal information including

Password, First Name, Last Name, Gender, Date of Birth, Email, Country

*RunnerID is non-editable by runner

View event record including

Check-in Time, Time to finish, Top speed

c. Register/ Deregister to an Event

Registration

Can view all the event types and able to register one of the event that will be hold on that year with a Race Kit.

De-registration

Can view all the event the runner was registered & able to deregister from it. (Only the future events.)

d. Payment

Can view the total amount (Event + selected Race Kit) they are required to pay.

*Simulate an online payment UI

4. Functions for the Volunteer *(25 marks)*

Done by

a. Register to System

Can create a new Volunteer account into the system with all necessary information.

Required information during registration

Password, First Name, Last Name, Gender, Email

* VolunteerID should be generate automatically by the system as Primary Key.
* VolunteerID will be shown when registration completed for login purpose.

b. View & Update Runner Event Records (for assigned runner)

Can view a list of assigned runners and add event record.

On & after the event day

Can help every runner to check-in to the current event with Runner ID & update their event records (Time to finish & Top speed).

c. Send Race Kit (for assigned runner)

List all unsent Race Kit of assigned runners and update the record when you released a set to them.

d. Manage Race Kit

Can create a new Race Kit record into the system with all necessary information.

Required information for creating a Race Kit

Race Kit Name, Description, Price, Photo, Event ID

Can delete a Race Kit record from the system.

5. Functions for Administrator *(25 marks)*

Done by

a. Create & Manage Events

Can create a new Event record into the system with all necessary information.

Required information for creating an Event

Event Name, Total Distance, Date of Event, Time Start, Price

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

Can modify & delete an Event record from the system.

*When an event is modified, a list of affected runners should be shown.

b. Manage Volunteer and Runner

Able to assign and unassigned a runner to volunteer.

As one runner must be assigned to a volunteer, a list of unassigned runners should be shown.

c. Create Charity Record

Can create a new Charity record into the system with all necessary information.

Required information for creating a Charity

Charity Name, Description, Website URL, Logo

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

d. Remove Sponsor Record

Can view & delete a sponsor record with no payment after the event day

6. Functions for Sponsors *(25 marks)*

Done by

a. Register to system

Can create a new Sponsor account into the system with all necessary information.

Required information during registration:

Password, First Name, Last Name, Company, Email

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

*SponsorID will be shown when registration completed for login purpose. _____

b. Sponsor Information

Can view and update his/her personal information.

*SponsorID is non-editable by sponsor _____

c. Sponsor a runner

View the all the runners record ordered by top speed and able to sponsor a runner with a charity and amount of money for the coming event. _____

d. Remove the Sponsor record

Able to cancel the sponsorship before the event starts and no payment. _____

7. Form your project group

Each student needs to form a project group, **the maximum number of students in each group is 4 (preferrably not less than 3)**. 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.

8. 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 = "";

9. Items to submit

a. A ***CD-ROM*** or ***DVD-ROM*** 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 Studio* 9 (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 9** http://www.techsmith.com/tutorial-camtasia-current.html :
Record Full Screen :
https://www.techsmith.com/tutorial-camtasia-9-3-record-edit-share.html (for Camtasia Studio 8)
Produce and Share an MP4 Video :
https://www.techsmith.com/tutorial-camtasia-9-3-produce-share.html (for Camtasia Studio 9)

10. 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

11. 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 behaviours (such as validate data in the form, highlight a table row with different background colour 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.)

12. 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