THE UNIVERSITY OF
NEWCASTLE
AUSTRALIA

School of Electrical Engineering and Computing

INFT3050 – Web Programming

Assignment 1

e-Commerce Shop Front Web Application

PART 2

Business Layer and Data Access Layer.

Due 11:59pm Friday the 7th of June, Week 13, Semester 1, 2019.
*This assignment is worth **40%** of the course assessment.*
This assignment can be completed individually or in groups of 2-3.

I. Introduction

INFT3050 is about Web Applications and their supporting infrastructure and technologies. Web applications come in many formats but unique to their appeal is the ability to run "anywhere" on "anything" as long as there is an Internet/Intranet connection and some form of web browser on the device being used. One of the most common web applications are those serving as an online shop (eCommerce functionality).

In a more generalised employment focus for new graduates, more often than not target skill sets gravitate towards Web Applications of a more business-orientated nature. Therefore, the core practical Assignment for INFT3050 will be the implementation of an eCommerce website with content/context decided by the student. The technological architecture, supporting software development tools and DB storage will be determined for you, consistent with being a first time employee in an organisation.

The Assignment consists of TWO parts, the second building upon the first. The focus of the PART 1 being on the Database to support the application and the User Interface (Layer). Part 2 of Assignment 2 focusses on the Business Logic (Layer) to make the Web Application functional and the Data Access Layer needed to send and retrieve information to the backend Database. The final total Assignment will be a fully functioning n-tier Web Application representing an Online Shop Front selling products/items determined (within reason) by the student.

II. Part 2 - Overview

Like the assignment that consists of two parts, this Part 2 of the assignment has two major components (like Part 1). The two components in Part 2 of the Assignment, form the completing layers to the first two completed in Part 1. That is, in Part 1 you completed the front end or User Layer and the back end or Data Layer. Part 2, this assignment is focussed on the two 'middle' layers, the Business Layer (BL) and the Data Access Layer (DAL). Hence, the overall web application will use an n-tier/n-layer architecture. In the Assignments instance, this will involve 3 tiers (the user browser tier, the web server tier, and database server tier) and 4 layers. As mentioned above, Part 1 of the assignment is the User Layer (UL) and the Data Layer (DL). Part 2 of the assignment is the Business Layer (BL) and the Data Access Layer (DAL). While some of you may already be familiar with other web application architectures such as MVC, this is assignment is TO BE IMPLEMENTED as an n-tier application. Having all assignments, use the same architecture ensures consistency across the marking guide and natural progression through the different parts of the assignment. The n-tier architecture also lends itself more readily to group work, and allows separation of tasks more readily.

The technologies defined in the next section are provided to you, and are the only ones to be used for this Assignment. As such, your Assignment project should be structured to reflect the n-tier architecture through the correct directory and project format, with each 'folder' representing one of the top three layers of the Application. For example, the final solution should at least have a UL, BL, and DAL folder where the respective code files are located. The DL is the actual DB that will be stored in a Relation Data Base Management System (RDBMS) and connected through a connection string in the Application configuration.

Technologies
- The IDE (Integrated Development Environment) you will be using is Microsoft Visual Studio. You will need to start by creating an 'Empty' web application for the whole Assignment. Do not use the integrated Administration/Security controls option.
- The RDMS (Relational Database Management System) will be Microsoft SQL Server. You will need to create a database and the export to a SQL Script file for submission again for Part 2 if you have changed the DB since Part 1.
- You will be coding in the C#.Net language for the code behind, Business Layer, and Data Access Layer files.
- You will be using ASP.Net to build the web application user-interface pages. These have an .aspx extension and will be used to populate the User Layer.
- In addition, you may use JavaScript, CSS, and HTML5 to enhance the user interface where appropriate, but the core UL technology is ASP.Net.
- We will provide a dummy payment system for you to use with your site

Application Architecture
- Business Layer – will be .cs files stored in the BL folder OR in a separate BL Project.
    1. The BL files are responsible for passing data between the User Layer and Data Access Layer.
    2. The BL is responsible for any business logic your application might need.
    3. You can chose to use a single .cs file that contains all of the Classes and Methods for the application OR have multiple .cs files containing specific Classes and Methods.
    4. All Classes/Methods in the Business Layer (BL) should be accessible only to the User Layer Calls.
- Data Access Layer will be .cs files stored in the DAL folder OR in a separate DAL Project

1. The DAL is responsible for passing data directly to and from the actual Database system.
2. It will contain either the raw SQL commands OR calls to Stored Procedures contained in the Database.
3. Only the BL has access to the Classes/Methods in the Data Access Layer. The Business Layer instantiates DAL objects through calls to the DAL classes/methods contained in the DAL .cs file(s).
4. Like the BL you can either have a single .cs file containing all the required classes and methods needed by the Web Application OR multiple .cs files with specific classes/methods.

- An articulated simple example of the Logic Flow for the Login Process (that will also be provided in codified form) is as follows
    1. You have a simple Login.aspx form containing two text box elements called txbUsername and txbPassword. In addition, a single button called btnSubmit.
    2. Users enter an email address and password into the Username and Password textboxes respectively and hit the Submit button.
    3. The Code Behind page in the User Layer, Login.aspx.cs retrieves the values in each respective textbox and assigns them to 2 variables, strUsername and strPassword.
    4. The Login.aspx.cs then calls a (for example Integer) method/class in the Business Layer passing the two variables (strUsername, strPassword).
    5. The Business Layer class double checks the Username is a valid email address, and the Password string also double checks it meets any password requirements.
    6. The Business layer hashes the password using md5 hashing to pass to the data access layer
    7. Once Error Checking complete the Business Layer code calls a method/class (again an Integer type) in the Data Access Layer passing the user name and hashed password.
    8. The Data Access Layer method retrieves the two variables and formulates number of SQL Commands (3 key cases to check). Including:
        - SELECT * from tblUsers where Username = Username and Password=strPasswordHash;
        - SELECT * from tblUsers where Username = Username;
    9. Based on the logic you code in the Data Access Layer you will have three conditions to pass back that you can assign respectively to values (0,1,2)
    10. The method now returns the Integer value to the Business Layer.
    11. The Business Layer performs any required additional business logic.
    12. The Business Layer then returns the Integer value to the Code Behind page in the User Layer.
    13. Code in the code behind page now manages one of three possible scenarios based on the returned integer. HINT: if it is a 2, for example, both Username and Password combination were found in the DB so login successful so Redirect to the main menu.

III. Assignment 1 – Part 2 Project Requirements

Below are the basic project requirements:

1. To design, develop, implement, and test all Business Layer Classes and Objects to complete the Assignment Web Application.

2. To design, develop, implement, and test all Data Access Layer Classes and Objects to complete the Assignment Web Application.

IV. <u>Knowledge that you need to have</u>

In order to successfully develop the set of assignments you need to at least have the following knowledge:
- C#.Net
- ASP.Net
- Microsoft SQL Server and T-SQL
- N-tier web architecture and n-layer web applications.
- Classes and Objects development using Object Orientated design principles.

V. <u>Assignment 1 – Part 2 Tasks, Tables, and Deliverables</u>

The key deliverables for Assignment 1 – Part 2 are:

1) The complete functioning Business Layer with all Class/Methods contained in their own .cs files (you can either store them within their own BL folder within a single Project and Solution OR you can store all the .cs files in a separate BL project contained in the single Assignment solution).
2) The complete functioning Data Access Layer with all Class/Methods contained in their own .cs files (you can either store them within their own DAL folder within a single Project and Solution OR you can store all the .cs files in a separate DAL project contained in the single Assignment solution).
3) The full Assignment Visual Studio solution/project with all files. This includes resubmitting of the User Layer (NOTE: you will not be marked again on any User Layer features, development or related content, this has already been marked in Part 1).
4) The full Database exported as a SQL Script file. There is a good chance you may make some changes to the DB and associated tables for Part 2 of the Assignment. As such, you will need to submit the Database again as a SQL script file.
5) A supporting document with details such as
   a. How to use the site if it is not self-explanatory.
   b. Login details for the Admin section and at least one User login for the Web Application itself.
   c. Clearly articulating and explaining any changes you have made to the DB or User Layer since Part 1 of the Assignment.
   d. References for any code/images or other digital content you have sourced that is not your own.

VI. <u>Submission Procedures</u>

You will need to deliver the following digital items for marking through Blackboard:
A. You must submit all of your files in a ZIP/RAR file called "CXXXXXXX_Assig1.zip" (or if in a team the student number will be of the person submitting the assignment, only 1 copy per group).
B. A digital cover page with the name's and student numbers of all group members.

NOTE: A full marking guide/rubric is also provided to guide you in this Assignment undertaking. It is on Blackboard.