Advanced Internet Technologies Project - 60 %

Lecturers: Michael Bradford, Anu Sahni, Vikas Sahni

Semester 1
2016 – 2017

ASP.Net Project

This assignment is worth **60%** of your overall mark.
Your project application code should be submitted via Moodle along with a project coversheet as an attachment. The project coversheet should give a brief overview of the project. All projects must be submitted on/before the deadline date. Projects submitted after the deadline will incur penalties.

Project Details

You are required to choose an area of interest to develop a web application using the ASP.NET MVC framework.

In this project, sample content will be generated for a particular topic. The content will be stored in a relational database. The application should exhibit functionality to allow some of the presented information to be edited. It should also be possible to update the database with the edited information and to delete information from the database.

Please see http://www.asp.net/ for examples of ASP.NET projects.

Tasks:

1. Create a project proposal document (approx. 2 pages). Your project proposal document should provide the following:

   a. An introduction to the project subject matter.
   b. The goals of the project.
   c. The strategy that will be employed to meet the stated project goals.
   d. The key characteristics of your project in terms of functionality.
   e. A list of the project deliverables that will be submitted on project completion.

   (5 marks)

2. Create an ASP.NET MVC Web Application Project. Modify the **_Layout.cshtml** view template to personalise your web-site.

   (5 marks)

3. Create a set of classes that will be responsible for implementing a Data Access Layer that will to communicate with a database you design.
   a. You should use the *Entity Framework* Object Relational Mapper (ORM) and a *CodeFirst* approach when creating entity classes.
   b. Your database should consist of at least 4 tables populated with data.
   c. Write a *Seed* method that the *Entity Framework* automatically calls after creating the database in order to populate it with test data.

   (25 marks)

4. Create a set of pages to provide a user interface for the display of information contained in your database. Also, create a set of pages to provide a user interface for the insertion, update, and deletion of information contained in your database.

   (25 marks)

5. Ajax functionality should be used to provide interactive effects on the webpages using data retrieved from your database (e.g., if the mouse hovers over a record ID then use Ajax to request detailed information from the webserver and display the summary data using an in page preview).

   (10 marks)

6. Create a web service that exposes an interface to allow querying of the data in your database. For example, the web service could accept a *table name* and a *record id* as parameters and then return the associated record from the database. Use of the web service should be integrated into your web application's functionality.

   (10 marks)

7. Modify the default Identity Provider functionality to store additional information for user accounts.

   (10 marks)

8. Deploy your application to Microsoft Azure or an IIS server installation.

   (10 marks)