



## FACULTY OF SCIENCE

### ACADEMY OF COMPUTER SCIENCE AND SOFTWARE ENGINEERING

**MODULE** IFM2B10 – INTERNET BASED ELECTRONIC COMMERCE

**CAMPUS** APK

**SEMESTER TEST 2**

**SSA** **SEPTEMBER 2015**

**PRACTICAL TEST**

**DATE:** 21/09/2015

**TIME:** 09:00 – 11:00

**ASSESSORS:**

MR SMA MAVEE

**INTERNAL MODERATOR**

MR D COTTERRELL

**DURATION:** 120 MINUTES

**MARKS:** 100

**NUMBER OF PAGES:** 3

#### INSTRUCTIONS:

- Ensure that you are seated at the computer that has been assigned to you.
- Create a folder in the **T:/ drive** with the following name format: CXXX\_YYYYYYYYY where X represents your computer number and Y represents your student number (e.g. student 201300001 sitting at computer number 10 will name his/her project Co10\_201300001). Save your project in this folder.
- Read through and complete the **Student Declaration** below when you are ready to submit your solution.

#### STUDENT DECLARATION – TO BE COMPLETED UPON SUBMISSION OF YOUR PROJECT

I, the student whose details appear below, declare that:

1. A **verified** zip file containing the full and final version of my ASP.NET project that I intend to submit for marking has been uploaded to Eve.
2. The abovementioned project has also been saved to **CD / USB** memory stick as backup.
3. I acknowledge that failure to properly verify that the solution submitted is the full and final solution I intend to submit can result in the forfeiture of marks.
4. **I am aware that non-compiling solutions will be capped at 40%.**
5. **I fully understand that only what is correctly submitted will be marked.**
6. **I fully understand that the marking rubric is also part of the question in the semester test.**
7. **I fully understand that not agreeing to the rules above will result in forfeiting the assessment opportunity.**

Student #:									
Signature:						PC #:			

Surname:						Initials:			
----------	--	--	--	--	--	-----------	--	--	--

Upload Method:	EVE	CD	USB	(Circle the applicable methods)
----------------	-----	----	-----	---------------------------------

Circle the selected programming language

VB	C#
----	----

RugbyWorld is a sports apparel retail company base in Johannesburg. They are the best known store that keeps all the latest gear rugby players need. Therefore right after establishing their website to sell their products online, they immediately gained a lot of business since more customers could be reached. Orders for rugby shirts, boots, balls, etc. shot through the selling and came from all corners of the world.

After having a great start to their online venture, the managers of the store then realised the importance of keeping their inventory records up to date. This is so that customers could always be informed of everything the store sold. They therefore decided to bring you in to the company to develop a web-based inventory management system for RugbyWorld. ASP.NET is the technology they would like you to use to build the web application.

The website will be used for internal operations only and will allow the identified set of users who will be given access to the system to log in so they can manage the products in RugbyWorld's inventory. The web application should also provide a list of all products RugbyWorld currently sells and should allow its registered users to add new products and edit information on products they already have in their inventory database.

**You have been provided with a basic ASP.NET project template which you will use to complete the test. On the project, you will find:**

- A .css file to style elements on the page and an **AppMedia** folder with images to be used,
- A .js file to link to the page to be created,
- A Master Page which **all** your web forms **MUST** use,
- A Secrecy class, with a HashPassword method which can be used to hash text using the SHA-1 algorithm, and
- A database with all the tables and information you will be using for the test. The database contains test data.

**Using the basic ASP.NET web application provided to you, extend it in order to do the following:**

- **Master Page**
  - Write HTML code to link the Master Page to the "MyCss.css" file in the Styles folder.
  - Write HTML code to link the Master Page to the "MyJS.js" file in the Scripts folder.
  - Add a navigation menu with links to the **Login Page** and the **Products List Page**.
  - Provide a Logout link that is only active if a user is logged in.
  - Write code in the "MyJS.js" file's **WriteTime()** method to write the current date and time in the Master Page's **timeDiv** tag.
- **Create a Login Page (for use by only the pre-loaded RugbyWorld users ONLY)**
  - The username and password provided by the must be **checked** against the login details stored in the database. After the authentication, users should be redirected to the **Products List Page**.
  - Display a welcome message after authentication. Welcome message may be put in the Master Page.
  - Use session variables to keep track of the authenticated user.
- **Create a dynamic Products List Page**
  - Display a dynamic list of all products available at RugbyWorld.
  - Display only the product's name, type, number of items available and short description.
  - **Set** this as the **Start Page** of your website.
  - Display the images allocated for each product on the list.
- **Create a dynamic Product Details Page**
  - Each product listed on your **Product List Page** must have a **link** that takes you to the **Product Details Page** and must make use of **URL parameters**.
  - This page should only display a selected product's full details (all of its fields, except the PK).
- **Create a Product Addition Page**
  - This page must **ONLY** be accessible to authenticated users.
  - On this page you allow authenticated users to **add** new products to the database.
  - Use validation controls to ensure all data to be collected is provided by the user.
  - Use the value "noimage.jpg" for the **ImageLocation** field in the **PRODUCT** table.
- **Create a Product Information Editing Page**
  - This page must **ONLY** be accessible to authenticated users.
  - On this page you allow authenticated users to edit information on a selected product entry.

<b>Student #:</b>									
<b>Signature:</b>						<b>PC #:</b>			

Mark Allocation								Marks	
<b>Master Page</b>								<b>12</b>	
- HTML code to link to CSS file								2	
- HTML code to link to JS file								2	
- Navigation menu with links to Login Page and Product List Page								3	
- Logout link. Only active if a user is logged in								2	
- WriteTime() Method								3	
<b>Login Page</b>								<b>12</b>	
- Authentication of User								8	
- Display of a welcome message								2	
- Create session variables to keep track of authenticated user								2	
<b>Dynamic Product List Page</b>								<b>20</b>	
- Retrieve and display list of products from database								13	
- Display of images								3	
- Creation of URL Parameters								3	
- Set as Start Page of the web application								1	
<b>Dynamic Product Details Page</b>								<b>17</b>	
- Use of URL Parameters								4	
- Retrieve and display full information on the selected product entry.								13	
<b>Product Addition Page</b>								<b>18</b>	
- Authentication Check (Only logged on users have access)								2	
- Form Design								3	
- Add new product entry to the database								10	
- Use of Validation controls (to enforce all details are entered)								3	
<b>Product Information Editing Page</b>								<b>21</b>	
- Authentication Check (Only logged on users have access)								2	
- Form Design								3	
- Retrieving and displaying selected product entry information from the database								6	
- Updating product information in the database								10	

<b>TOTAL</b>								<b>100</b>	
<i><b>Solution Does not Compile – Cap at 40%</b></i>								<b>100</b>	

<b>For office use only</b>	<b>SMA Mavee</b>	<b>Date:</b>	<b>D Cotterrell</b>	<b>Date:</b>
----------------------------	------------------	--------------	---------------------	--------------