

Coursework Assessment: Design and Implement an Advanced Web Application

This coursework comprises 100% of the Final Module Mark. Release date for this Assessment Exercise is Monday, 18th March 2024, 00:00 midnight and the submission deadline for this work is Tuesday, 23rd April 2024, 13:00.

I. Introduction:

Many food companies are trying to reduce their cost by managing their online food store to reduce their physical storage requirements, manage inventories efficiently and serve their customers better. Customers can reduce their visits to their shops, browse the food products and make their ordering/choices efficiently.

In this coursework assessment, you will need to develop a system to be used by customers/buyers and maintained by administrators (food company staff). The website should be written using standards compliant HTML5 and CSS3 code and the use of templates and frameworks such as Bootstrap are allowed (but must be referenced in the code and report). However, be aware that marks can only be awarded for your own modifications to templates and frameworks. Your system should be responsive, adjusting intelligently to different display capabilities. Your work will be evaluated using the version of the Google Chrome Web browser in lab PCs available inside lab CR010 (your usual practical classes). Remember that the demonstrators cannot write your assessment for you, but they can answer technical questions and point you in the right direction. Make sure to ask them/myself if you have any questions or problems during your lab sessions.

Marking Scheme: You will need to develop the following functionalities, consider designing aspects, coding quality, bug free, consistency of presentation, and descriptive report as outlined below. The assessment will consider both comments in the codes and accompanying report. You will be called upon to demonstrate your work at a specific date and time. (a) Functionality Aspects and Implementation (65%), (b) Designing and Coding Aspects (15%) and (c) Description and Working Procedure Report (20%). Detailed mark distributions for each section are shown below as a percentage (%).

II. Functionality Aspects and Implementation (This section total 65%):

1. The user should be able open the **food company webpage** and browse all the available food products purchase options. For example, the first drop-down menu should show two items: *Burger and Sandwich*. When:

(1) *Burger* is selected, it should populate second drop-down menu providing options to select one from a few available foods, such as:

- (1)(a) Chicken,
- (1)(b) Vegetable and
- (1)(c) Fish,

and their corresponding pictures (any flavour/type is fine).

School of Computer Science and Mathematics
CSC-30025: Advance Web Technologies
ASSESSED EXERCISE

Similarly, when:

(2) *Sandwich* is selected, it should populate second drop-down menu providing options to select one from a few categories available, such as:

- (2)(a) BBQ,
- (2)(b) Chicken and
- (2)(c) Falafel,
- (2)(d) Fish.

and their corresponding pictures (any flavour/type is fine).

All the items are priced differently. You should use efficient asynchronous JavaScript and XML (AJAX) for building the food company webpage. Their information names, images and prices should be stored in **table(s) belonging to one MySQL database**. The website viewed by the customer should have **register and login links/buttons** for the user to register and login (more details in II.3 and II.5 below). **(10%)**

2. While designing the food company website, search engine optimization aspects should be followed, where possible. **(5%)**

3. On the customer **registration page**, the user should register securely with Name, Phone number, Email ID (unique) and a Password. Validation of the email ID (such as abc@xyz.com) and phone number (such as XXXXXXXXXX (10 digits)) are required. Email ID is unique in the table (no two persons should have the same email ID). Security aspects such as: Cross-site scripting (XSS) attack, HTML entities and SQL injections should be taken care of. **(10%)**

4. During the time of registration, all these four personal information (Name, Phone number, Email ID and password) should be stored in the **another MySQL table** (database could be the same but a different table). **(5%)**

5. On the **login page**, existing users should be able to login with email ID and password and logout securely; authentication details are kept using the session variable. Login is also validated with a **CAPTCHA** (completely automated public turing test to tell computers and humans apart) as a security measure for response authentication. You could use the existing captcha images provided in the practical/lecture class (also attached in the assignment on KLE) or make your own images. **(10%)**

6. Form processing (**while registering**) should validate all fields as the user performs the keystroke (i.e. live validation), using the React framework. For example, the name should only contain letters, the phone number should only have numbers, valid email address, etc. Any mistakes or errors, should prompt (an appropriate message to) the user immediately (while the user is typing). **(10%)**

7. Once successfully registered/logged in, the user should receive an appropriate message and be redirected to the food company website (as described in II.1). Where the user should be able to select at least 1 product item and receive its pricing on the screen. Note that on the food company webpage II.1, the user would be able to view the product items and their prices, but they won't be able to order. However, when successfully logged in / registered, they should be able to both view and **submit an order**. The ordered item

School of Computer Science and Mathematics
CSC-30025: Advance Web Technologies
ASSESSED EXERCISE

should be stored in a **new MySQL table**, (can belong to the same database) with customer name, email ID, phone number, ordered item and their price. **(10%)**

8. Finally, develop a REpresentational State Transfer (RESTful) web service for the Manager in the food company to view the order placed item by item online by sending the order ID (over the web API, primary key in the table) stored in the ordered items table. **(5%)**

III. Design and Coding Aspects (This section total 15%):

1. Design: Marks will be awarded for the navigation and user interface design and how they conform to standard usability and accessibility criteria. **(10%)**

2. Code Quality: Marks will awarded for appropriate code design, field types, commenting, modularization, bug free and quality. **(5%)**

IV. Description and Working Procedure Report (This section total 20%):

The report should comprise of:

1. Introduction: Information about the architecture of the site and an overview of the different components of the project. Include diagrams and figures where necessary (hand drawn are fine, include them in the document as pictures with captions). **(3%)**

2. Developed Architecture, Implementation Details and System Inputs and Outputs: A detailed description of the working procedure of your designed website (similar to a short software product manual). For example, mention your starting webpage, ways to view the information and then browse to which page and then to which page. This should have the working guidance (user guide) of your software. **(15%)**

3. Suggested Improvements and Conclusions: Suggested ways that your work could be improved. **(2%)**

A task template is shown in Table 1 (to be included in the beginning of your report in addition to the above points IV.1 to IV.3):

Table 1. Task template. Include this table in your report and provide details about each task. Comment on what you achieved, where you experienced problems and any parts you were unable to complete.

Task	Comments
1. Food company webpage	
2. Submit SQL files to generate the tables in the food company	
3. Registration page	
4. Login page (mention whether	

School of Computer Science and Mathematics
CSC-30025: Advance Web Technologies
ASSESSED EXERCISE

with/without Captcha)	
5. Security aspects (preventing attacks) of the Registration and Login page	
6. Search engine optimization aspects in the food company webpage	
7. User Registration validation using React framework	
8. Data management	a) Customer registration table:
	b) Customer ordered table:
9. RESTful webservice for the food company Manager	
10. Any other information (remarks)	

You should focus your writing on the industry adaptation of web technologies and must not exceed 2000 words, including the title, references, figure, table captions and contents. **The student id, the login details, a link to the live version of the website and the number of words MUST be provided at the beginning of the report.** Reports will be marked according to the university's marking scheme, available at <https://www.keele.ac.uk/sas/academicservices/assessment/markingscheme>.

Sections: II (65%) + III (15%) + IV (20%) = 100%.

What resources might I use to get started?

Ans: Laboratory CR010 PCs, University network and software.

Module Leader & Contact: Bappaditya Mandal (Email: b.mandal@keele.ac.uk)

Office: Colin Reeves CR36, Hours: Ask during any of the practical classes in CR010 or by appointment via email or MS Teams chat.

Module Learning Outcomes Assessed
1. Design and implement advanced modular multi-tier web applications.
2. Evaluate techniques to create distributed web applications.
3. Assess issues related to software architecture and web design patterns.
4. Apply appropriate user interface and user experience design techniques.
5. Assess and apply web security approaches.

V. Things to submit (make sure that you)

1. Make a .txt text file named with your 8-digit student ID, for example “02915643.txt”, put the web link to your webpage. For example:
http://www.teach.scam.keele.ac.uk/prin/your_username/folderName/
(Instead of <http://localhost/02915643/>)
and save this link in the 02915643.txt text file. During your demonstration, the assessor would open this text file, go to the provided link, check your website and evaluate your coursework assessment. Before you submit, **you must verify that your provided link is working** within the University network (laboratory PCs).
2. You should use the assignment drop-box for this module on the KLE. Attach your written report as a PDF document. The zip file will include all the source code, above text file and the report in PDF file (e.g. 02915643.pdf). Please note that .zip is the only permitted format. Other archive formats such as .rar and .tgz are NOT acceptable and will not be marked. Name the zip file according to your 8-digit student ID (e.g. 02915643.zip). Please attach a **single compressed (.zip) archive file** of your website source code files (PHP, HTML, SQL and others), text file and report (.pdf file) and upload to the appropriate dropbox in the KLE before the date and time shown in the beginning of this document.

VI. Advice and Tips

1. This assignment should be your own work and may need to be completed in your own time: just like the independent study time that you spend working on items of coursework for your other taught modules back at the University. You should not seek or request assistance from any code-sharing or code-development websites or companies.
2. Of course you are encouraged to refer to, and reuse as appropriate, any of the course materials that you have been provided with or developed earlier in your degree studies. However, if you make use of the published work of other scholars, authors or software developers you should ensure that this usage is clearly stated and properly acknowledged in your submitted material. You are responsible for ensuring that any discussion of ideas with other students does not transgress the guidelines provided in the School Handbook or the University’s guidelines. In particular, you should neither supply material to other students, or request it from others. Remember that you are responsible for keeping your work securely.
3. Help & Guidance: Remember that, while this is an individual element of assessment, you are strongly encouraged to obtain any technical support, help and guidance through the usual channels including: your CSC-30025 lecturers, tutors, teaching assistants and demonstrators. **Absolutely no penalty is incurred by doing so.**
4. Your implementation should be based around PHP, HTML5, JavaScript, CSS3, MySQL and the React JavaScript library. You should use the enhanced interactivity and advanced AJAX facilities.
5. When designing your website, take time to sketch out your ambition on paper first. Draw the pages and note how they link to each other.

School of Computer Science and Mathematics
CSC-30025: Advance Web Technologies
ASSESSED EXERCISE

6. Remember and apply the principles for interaction design that were presented in your previous years.
7. If necessary, you could use the images under creative commons licensed images (details here: https://en.wikipedia.org/wiki/Creative_Commons_license). Some example sample images can be searched here:
<https://wordpress.org/openverse/?referrer=creativecommons.org>.
8. Build the structure of your site first, then go back and embellish the content and presentation later.
9. Think carefully about your chosen approach. Make sure it is open enough to allow you to develop an interesting argument in the report, but not so open that you can get lost trying to cover the details.
10. If you are citing the writing of others, try to capture why they made the point you are citing, as well as the simple fact that they did so.
11. Always proof-read your work carefully to check for spelling mistakes, grammar errors, and confusing ordering. Make sure your provided link, for example:
http://www.teach.scam.keele.ac.uk/prin/your_username/folderName/
and save this link in the 02915643.txt text file is working. Give it a careful once-over before submission to ensure everything is in order.
12. Always revise all your codes and their functionalities. When you are happy about your developed codes and documentation then you should submit before the deadline.