

# SOFT7008 A#2 Design, develop and implement a basic e-commerce store.

---

Dr Ruairi O'Reilly

April 25, 2017

## 1 THE BRIEF

You have being approached by a small business to design an e-commerce store to sell the goods and services offered by that the business. Choose a business area e.g. retail, hospitality, catering and develop a web app that fulfills the structural, styling and technical requirements outlined below.

Provide functionality to list and sell products. There are two types of user, the customer and the store owner. The customer should be able to select a desired product, specify the desired quantity of the product and add the product(s) to their basket upon hitting an order button. The order should be processed by the e-commerce store when the user hits a "Buy now" button. The owner should be able to add new products to his/her store and have them persisted to a database.

### 1.1 ASSIGNMENT DETAILS

Note: the assignment is worth 60% and is due Semester end - Sunday the 14th of May @ 23:59.

### 1.2 EXPECTED INPUT & OUTPUT OF THE SITE

You have received the following requirements from the store owner. The owner wants a page that allows customers to view the available service/products, add that product service to a cart, and finally click a buy now button that purchases the contents of that cart.

To be included in the site:

- A logo, positioned at the top of each page.

- A horizontal navigation bar, directly beneath the logo including nav links to the different pages of the store.
- Product Catalog - Where the user can select from a variety of choices in relation to the product/service. Each product/service listed should include the following fields:
  - Product Name: Textual description of the product/service
  - Price: Cost of 1 unit of product/service
  - Picture: An image depicting the product/service
  - Quantity: A depiction of how many units he/she is interested in
- An “Add to Order” button - When this is selected the products currently selected (based on quantity larger than 0) should be added to the users cart.
- An order review panel is depicted below the Catalog showing all the products that have been added to the cart and a subtotal for said products/services dynamically calculated.
- A “Buy now” button that is clickable by the user, this will process the order and persist it to the db.
- The footer should depict the business address on the far left and the contact details (phone, e-mail) on the far right. Default applications should be launched for the handling of both phone and e-mail.

### 1.3 ASSESSMENT

- Part A - Document all the javascript code in your app.js file using comments and a separate code description document. You may use a flow chart, pseudo code, diagrams etc to describe how
- Part B - You will be using mysql. A script called productDB.sql will be provided for you containing the products database setup. (We'll run through this process Thur evening and a step-by-step guide will be provided thereafter)
- Part C - Write the code for a creating the relevant pages - break down the process into what is dynamic and what is not. (Basic functions for inserting/retrieving from mysql will be provided to you). Tip: Separate header/footer html and code that up asap, draw a rough wireframe if how you'll navigate the site.
- Part D - Dynamic elements - adding product, display products, add to cart, buy now - break these down and simplify as much as possible
- Part E - Combine the items in the cart if the user orders a product more than once. You need to modify the add endpoint to achieve this.

The assignment will be assessed based on the structure, styling and interactivity of the site. Other important features include good coding conventions and the degree to which the requirements are fulfilled. It is also important that you are able to articulate an understanding of how you have solved each requirement.

The assignment folder is to be [zipped](#) and submitted via blackboard by the due date. Required documents:

- HTML Files (header.html, footer.html)
- CSS File (style.css)
- JavaScript File (app.js, package.json)
- Folder (img) and any images