CIT, DEPARTMENT OF COMPUTING

Design, develop and implement an e-commerce store for the TATE Art Gallery.

Dr Ruairi O'Reilly

November 1, 2017

1 THE BRIEF

You have being approached by the TATE gallery (http://www.tate.org.uk/)to setup an e-commerce store to sell prints of their art collection. It should present dynamic views of the collection curated by the Tate gallery. That is, generated from the collection of data provided, that will present information based on the section of the site the user is interacting with.

- There are two types of user, the customer and the curator(store admin).
- Provide functionality to list and sell products. (The owner should be able to add new products to his/her store and have them persisted to a database)
- The customer should be able to browse the available prints by artist, artwork and genre. (Hint: think routes)
- He/She should be able to specify the desired quantity of prints for a selected artwork and add the product(s) to their basket upon hitting an "Add to Order" button.
- The order should be processed by the e-commerce store when the user hits a "Order now" button.

1.1 ASSIGNMENT DETAILS

Note: the assignment is worth 30% and is due by the end of Week 12 - Sunday the 3rd of Dec @ 23:59.

1.2 Input data

The dataset for the Tate gallery collection is available on github in JSON which is handy if you wish to use Mongodb:

`https://github.com/tategallery/collection`

The dataset has also being converted to a mySQL format that will be compatible with sqlite.

1.3 Expected Input & Output of the site

You have received the following requirements from the galleries curator:
The owner wants a page that allows customers to view the available works, add that artwork 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. (image)
- 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.

1.4 Assessment

Note: You are required to submit a 5 minute screencast in which you show all required functionality working, or not, and articulate an understanding of how the underlying technologies operate - This is mandatory.

- 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 - 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 C - Dynamic elements - adding product, display products, add to cart, buy now - break these down and simplify as much as possible

- Part D - 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.