CS602\_Term\_Project\_Lowenthal

Author: Jacob Lowenthal

Title: Shopping Cart Application

The purpose of this application is to mimic a shopping cart application. The customer can search for products, add products to a cart, and checkout. The admin can manage the inventory of an online store by adding products, updating current products, deleting products, and searching for products.

In order to run this application, you need the following programs and languages installed on your computer:

-MongoDB

-Node.JS

-CURL

First, run npm install to install the node modules in the project directory.

In addition, there are three other Node dependencies installed in this application: Express.JS, Express Handlebars, and Mongoose. If they are not already installed, then use this command:

npm install –save “libraryName”

The credentials for the database name, username, password, host, and port are located in the credentials.js file. If you wish to connect to your existing database, update these credentials with your own. If you wish to create a database with credentials listed in that file, then credentials.js does not need to be updated. The collections in MongoDB will show appear under the following title:

"productModelJL"

If you use a MongoDB font-end application like “Robo-3T”, then the collections will appear as “productModelJLs”. Once you have the dependencies installed, start the MongoDB.

Then, in your terminal, navigate to my term project folder, and start the server application with the following command:

node serverScript.js

If you need to insert some dummy data into the database to test the functionality of the application, run test.js in a separate terminal window with the following command while both MongoDB and serverScript.js are running:

node test.js

At this point, you should be able to open up your browser to navigate through the application.

The search functionality supports requests for XML and JSON, and will output search results accordingly. You can test the customer search functionality with the CURL utility using the following commands:

//test search\_by\_name for json

(filepath)curl.exe -H "Accept:application/json" "http://localhost:3000/customer/search\_by\_name/" --request POST --data "search\_by\_name=drum%20set&submit=search"

//test search\_by\_description for json

(filepath)curl.exe -H "Accept:application/json" "http://localhost:3000/customer/search\_by\_description/" --request POST --data "search\_by\_description=guitar&submit=search"

//test search\_by\_name for xml

(filepath)curl.exe -H "Accept:application/xml" "http://localhost:3000/customer/search\_by\_name/" --request POST --data "search\_by\_name=drum%20set&submit=search"

//test search\_by\_description for xml

(filepath)curl.exe -H "Accept:application/xml" "http://localhost:3000/customer/search\_by\_description/" --request POST --data "search\_by\_description=guitar&submit=search"

If you wish to test the search functionality for the admin user change "customer" to "admin" in the URL path for each CURL command. Place the file path on your machine where curl.exe is located in place of (filepath) for each command.