Memoria Entrega 1 Seminario SAD

Trabajo realizado por:

* Sylwia Zoń
* Jorge Andreu Royo

El código mencionado en esta memoria puede ser accedido desde el siguiente repositorio: <https://github.com/joranro1997/SAD>

En la presente memoria se destacan detalles sobre la realización del código solicitado para la entrega de esta tarea.

El código de las entregas de esta tarea ha sido probado bajo la versión 16.10.0 de node y la versión 5.0.3 de mongodb.

Se ha hecho uso de npm para la instalación de mongo.

En respuesta al paso 1 del enunciado de la actividad, se ha creado una clase en el archivo ‘CartMaker.js’ que integra todas las operaciones básicas definidas en dicho enunciado.

La clase creada en este código hace uso de un array como forma de almacenar los productos del carro.

Esta clase se exporta como un módulo que posteriormente será utilizado en un script de pruebas.

El script ‘testScript.js’ presenta un código sencillo en el que se importa el módulo creado en el paso 1, se crea una instancia de la clase exportada en el paso 1 y se hacen llamadas a funciones de dicha clase para comprobar que funciona como se espera.

Para el paso 3 se ha realizado el módulo ‘manipulateStock.js’. Este módulo define una clase que hace uso de mongo para funcionar como base de datos donde se almacenan los contenidos del carrito.

Esta clase crea un cliente de mongo y al ejecutar la función ‘interactWithDb’ conecta dicho cliente a la url especificada en el constructor de la clase. Esta función también crea una colección para los productos del carrito si dicha colección aun no ha sido creada. Esta función será llamada cada vez que se quiera interactuar con la base de datos en cualquier medida, y se le pasa por parámetros la función que se desea ejecutar. Se hace uso de promesas en la llamada a diversas funciones de mongo.

Las funciones ‘insertGoods’ y ‘checkGoods’ ambas compruebas si existe una entrada en la base de datos y actúan en base al resultado de la consulta. ‘insertGoods’ comprueba si ya se ha poblado la base de datos inicialmente mientras que ‘checkGoods’ busca una entrada en particular pasada por parámetros.

Ambas funciones reciben en su llamada desde ‘interactWithDb’ la base de datos, variable ‘db’ y el cliente de mongo, variable ‘client’. En ambos casos se utiliza ‘db’ para instanciar la colección y ‘client’ para cerrar la conexión al finalizar la función.

Las funciones ‘addItems’ y ‘isInStock’ hacen las llamadas a ‘interactWithDb’ pasando las funciones pertinentes.

Para el paso 4, al igual que en el paso 2, se ha desarrollado un script, ‘checkStock.js’ que importa la clase creada en el paso 3, crea una instancia de dicha clase pasando una url local y realiza varias llamadas a funciones para probar que funcionan.

Para probar la ejecución de este código es necesario tener el servicio de mongodb en marcha en el equipo de pruebas tras haber instalado mongo con un ‘npm init’ seguido de ‘npm install mongodb’.