

Tarea Programada #1

- La tarea debe entregarse al profesor por medio del TEC Digital antes del día y la hora convenida.
- La tarea debe contener lo siguiente:
 - a. Fuentes, todo el código necesario para ejecutar la tarea. El código debe estar debidamente documentado.
 - b. Documentación, incluyendo al menos:
 - i. Documentación en el código
 - ii. Explicación del diseño y Arquitectura
 - iii. Manual de Usuario
 - c. Todo debe estar contenido en un Zip file que sea <INICIAL><APELLIDO>, ejemplo EHERNANDEZ.zip, si hay más de un miembro en el equipo, separen los nombres con UNDERSCORE.
- Toda tarea debe ser defendida ante el profesor, de tal manera todos los estudiantes deben poder explicar la solución satisfactoriamente.
- ¡Buena Suerte!

A Evaluar	Puntos	Nota
Documentación	5	
Productos (Incluyendo inventario e imágenes)	15	
Usuarios (Incluyendo las relaciones)	15	
Shopping cart y órdenes de compra	10	
Recomendaciones	15	
Comentarios y Calificaciones	15	
Uso de todas las bases de datos	25	
Total	100	
Videos y Fotos en los comentarios	5	
Recomendaciones con Grafos	10	
Total	115	

Mini Shopping Site

La tarea consiste en general un pequeño sitio de compras, donde se puedan registrar usuarios, mantener un inventario de productos con fotos, se puedan agregar productos a un cart y además, se puedan hacer la compras.

Cada parte del sistema se va a guardar en una base de datos distinta, de acuerdo con la información.

Los requerimientos serán los siguientes:

1. Se deben poder crear usuarios, tanto para los clientes como para los administradores del sitio.
2. Los administradores deben poder insertar nuevos productos, con sus imágenes al inventario del sistema.
3. Los usuarios deben poder tener un cart, donde agregar los productos que quieren comprar, este cart debe ser editable.
4. Los usuarios deben poder calificar de 1 a 5 los productos después de haberlos comprados.
5. Los usuarios deben poder hacer comentarios sobre los productos aún si no han comprado los productos.
6. El sistema debe poder sugerir a los usuarios productos que otros usuarios han comprado junto con el producto que se está revisando.
7. Los usuarios deben poder comprar los productos en el cart y debe poder ver las ordenes que ha comprado.
8. Las ordenes deben ser usadas para poder hacer las recomendaciones.
9. Los usuarios deben poder seguir las compras de otros usuarios si así lo deciden.
10. Todas las imágenes deben ser guardadas en MongoDB.
11. El cart debe ser guardado en Redis
12. Para las relaciones entre los usuarios se debe usar Neo4j.
13. Los comentarios se deben guardar en Cassandra.
14. Las órdenes de compra se deben de guardar en MySQL/MariaDB.

Para resto de los requerimientos como inventario de productos, calificaciones, recomendaciones, etc., deben de decidir en cuál BD se van a guardar y deben justificarlo en el documento de arquitectura.

En cuanto al lenguaje de programación debe ser .NET Framework 4.5, Visual Studio 2015, se debe usar MVC para la arquitectura. El Front End debe ser Web, no hay limitación en el uso de Javascript o frameworks para el front end, siempre que estos sean libres (open source) y estén en Javascript.

Puntos Extra

1. Que los usuarios en sus comentarios puedan adjuntar Fotos y Videos, estos deben guardarse en MongoDB y deben poder ver ambos en el front end. (5 puntos)
2. Replicar las órdenes de compra a Neo4j para poder hacer mejores recomendaciones relacionando usuarios, órdenes y productos. (10 puntos)