

Cargo:	Docente		
Nombre:	M.Sc. David Fabián Cevallos Salas		
Asignatura:	Programación Web		
Carrera:	Marketing Digital y Comercio Electrónico	Nivel:	Primer nivel
Estudiante:			

PROYECTO DE FIN DE MÓDULO

DISEÑO DE UN SISTEMA INFORMÁTICO PARA GESTIÓN DE UNA TIENDA VIRTUAL EN LÍNEA

1. Objetivos

1.1. Objetivo general

Diseñar un sistema informático para la gestión de una tienda virtual en línea que permita dar servicio a varios clientes.

1.2. Objetivos específicos

- Identificar las necesidades y funcionalidades esenciales para el diseño de un sistema de gestión de tienda en línea.
- Definir el pseudocódigo de software necesario para soportar una tienda virtual funcional.
- Establecer los flujos de procesos clave, como la navegación de productos, compra y pago, y la gestión de inventarios.

2. Antecedentes/Escenario

La venta en línea ha crecido significativamente debido al incremento de las compras electrónicas y la facilidad de acceder a productos desde cualquier lugar. Las tiendas virtuales deben ser accesibles, fáciles de navegar y ofrecer un proceso de compra seguro y eficiente. Un sistema informático para gestionar una tienda virtual debe cubrir áreas como:

- Gestión de inventarios (productos, categorías, stock).
- Procesos de compras y pagos en línea.
- Gestión de clientes y sus historiales.
- Administración de pedidos y envíos.
- Seguridad en la transacción de datos sensibles.

El diseño de este sistema se debe centrar en la funcionalidad, la seguridad, y la escalabilidad, asegurando que la tienda virtual sea capaz de soportar un número elevado de usuarios y transacciones.



3. Recursos necesarios

3.1. Conocimientos previos:

- Comprensión básica de algoritmos y estructuras de datos.
- Conocimiento sobre cómo funcionan las tiendas virtuales y los procesos implicados (gestión de productos, pedidos, pagos).
- Familiaridad con la lógica de programación, como variables, ciclos (bucles), condicionales, y funciones.

3.2. Herramientas para diseñar pseudocódigos:

- **Herramientas de texto:** Simplemente un editor de texto (Notepad, Word, Google Docs) para escribir el pseudocódigo.
- **Software de diagramación** (opcional): Herramientas como Lucidchart o draw.io para representar diagramas de flujo que acompañen al pseudocódigo.
- **Lenguajes de programación** (para pruebas de implementación posteriores): Familiarizarse con lenguajes como Python, Java o JavaScript para probar los pseudocódigos.

3.3. Ejemplos de algoritmos estándar:

- Conocer cómo escribir pseudocódigos para procesos de compra, gestión de inventarios, validación de pagos y generación de pedidos

4. Planteamiento del problema

Diseñar un programa que sugiera una solución para el siguiente escenario.

Una tienda en línea dispone de un catálogo de productos (Usted debe definir la cantidad de productos que desea para su catálogo). Se desea diseñar un programa que realice las siguientes acciones:

- Presente un menú con el catálogo de productos, cada uno numerado de 1 a n (donde n es la cantidad total de productos).
- Lea desde el teclado el número de producto que el usuario desea visualizar.
- Despliegue para el producto seleccionado la información del producto. Esta información puede ser una descripción corta del producto.

Presentar:

- Enunciado
- Análisis
- Pseudocódigo





- Prueba de escritorio
- (Opcional) Implementación en un lenguaje de programación

5. Pasos por realizar

5.1. Dar solución al problema planteada.

5.2. Elaborar el informe del proyecto **usando el formato determinado por el Instituto Tecnológico Quito** indicando los detalles de diseño e implementación. El documento a entregarse debe:

Poseer un índice de contenido, figuras y tablas debidamente numerado.

El texto debe estar debidamente justificado.

Cada página debe estar debidamente numerada.

Toda figurada y tabla debe poseer numeración y mencionarse en el texto.

¡Cuidado con el plagio!. Debe citar las fuentes de información que empleó.

6. Desarrollo

