

Programación II

Final

Profesor: Santiago Gallino.

Importante:

Este trabajo puede, y se recomienda, ser una continuación del 2do TP, habiendo corregido las correcciones indicadas por el docente.

Consigna:

Realizar un sitio tipo e-commerce que cumpla con los siguientes requerimientos:

- Contener todas las secciones listadas más abajo, que incluyen las del segundo TP.
- Tener al menos 2 (dos) roles de usuarios: Administrador y común.
- Implementar un carrito de compras para que los usuarios comunes puedan agregar/quitar productos, y realizar una "compra" (simulada, no se requiere una pasarela de pago).
- Autenticación de usuarios administradores para acceder a un panel de control.
- El panel de control debe incluir la administración ABM (Altas-Bajas-Modificaciones) de dichos datos.
- El panel de control debe incluir un listado de usuarios registrados, y la posibilidad de ver las compras realizadas por cada uno.
- Implementar inclusión de secciones vía **include/require**, para evitar duplicar HTML.
- Manejo de usuarios e "ítems" a través de la base de datos y autenticación mediante sesiones.
- **Utilizar una estructura de HTML5 semántica y correcta.** Fallos *graves* en este punto **puede ser motivo de desaprobación**, o grandes penalizaciones a la nota final.

- Presentar estilización en CSS con un diseño acorde a la temática abordada (no se aceptarán trabajos en HTML pelado). Puede usarse frameworks de CSS como Bootstrap o Tailwind de base, pero debe tener estilos propios escritos por el/la alumno/a.
- Todas las rutas deben ser relativas, o absolutas calculadas dinámicamente. **No** debe haber rutas que apunten a una carpeta local de la computadora del alumno/a.

Sitio

Como se estipuló, el sitio debe constar de:

- "Home": Pantalla introductoria que contenga, al menos, de lo que se trata el sitio, de manera que sea informativo para un nuevo visitante.
- "Listado": Pantalla donde se listen los "ítems" que se estén ofreciendo, detallando lo más importante. Cada ítem debe poder clickearse para entrar a ver más contenido del mismo. Estos ítems deben obtenerse de la base de datos.
- "Detalle": Pantalla donde se muestren los detalles expandidos del ítem clickeado en el listado. En esta sección un usuario autenticado debe poder agregar el producto a su carrito de compras.
- "Registro": Pantalla donde un usuario pueda crear una cuenta (como usuario común).
- "Iniciar Sesión": Pantalla donde un usuario pueda autenticarse en el sitio.
- "Perfil": Pantalla que muestre los datos del usuario autenticado.
- "Carrito": Pantalla donde el usuario pueda ver su carrito, quitar ítems, y "finalizar" la compra. La finalización de la compra solo debe guardar en la base de datos la compra realizada (los productos incluidos, total, fecha, etc), y vaciar el carrito. No se requiere de una pasarela de pagos.
- "Contacto": Formulario de contacto donde los visitantes del sitio puedan completar para obtener más información sobre algún tema.

- "Panel": Formulario de autenticación para el "administrador" que de acceso al panel de administración del sitio.
- "ABM": El panel debe tener una administración de los "ítems" del proyecto. Deben poder hacerse altas, bajas y modificaciones de los datos.
- "Listado de usuarios: El panel debe tener, al menos, un listado de usuarios donde pueda entrar a ver las compras realizadas por cada uno.

Puntos extras

Para alcanzar una nota de 10 (diez), debe incluirse:

- Buscador y paginación de resultados en los respectivos listados.
- Posibilidad de manejar la cantidad de cada producto en el carrito de compras.
- Listado de las compras realizadas por un usuario en su perfil.

Base de datos

La base de datos debe llamarse **dw3_apellido_nombre** (ej: dw3_gallino_santiago), incluir el DER de la misma, y el SQL para su importación con datos ya cargados. En caso de estar hecho en grupo, la base debe tener el formato de **dw3_apellido1_apellido2** (ej: dw3_gallino_noto).

Los datos deben ser "reales", no texto simulado ("Lorem ipsum" o similar).

Debe estar normalizada.

Debe contar con tablas para:

- usuarios
- roles

- compras realizadas
- "items": que incluya al menos 5 campos.
- *Al menos* una tabla adicional relacionada con la tabla de "items".

Detalle extendido

HTML+CSS

El código **HTML** debe presentar una estructura **semánticamente correcta**, aprovechando las etiquetas de HTML5 creadas para tal fin. Esto no significa que tiene que haber ejemplos de uso de **todas** las etiquetas que hay, sino que deben usarse las que correspondan para el contenido que se presente. Adicionalmente, **no** significa tampoco que **esté prohibido** el uso de divs, sino que deben utilizarse para estilización o cuando no haya una etiqueta semántica que se ajuste mejor a la situación.

El sitio debe presentar **estilización en CSS**, con un diseño que esté acorde al contenido y target demográfico del sitio. Pueden utilizar frameworks de CSS para ayudarse (como Bootstrap o Tailwind), pero debe incluir estilización personalizada por parte del o la alumno/a.

PHP

- El sitio debe implementar una carga de secciones dinámica en un template de base vía parámetros por GET, como se vió en la cursada.
- Debe tener conexión a la base de datos vía la clase PDO. Toda variable proveniente del usuario que se utilice en el query *debe* ser incluida con holders, para prevenir vulnerabilidades de inyección SQL. El uso de variables (cuyos valores provienen del cliente) *dentro* del query **es motivo de desaprobación**.
- Debe haber ejemplos de consultas SELECT, INSERT, UPDATE y DELETE (ABM y listado básicos).

- **Debe** haber uso de clases, incluyendo al menos:
 - Para estructurar los datos utilizados en la aplicación, como ser para los productos o usuarios (tipo la clase Noticia de la cursada).
 - Conexión a la base de datos.
 - Carrito de compras.
 - Autenticación.

Se evaluará y tendrá impacto en la nota también:

- Complejidad de la tarea realizada.
- Usabilidad y accesibilidad del sitio.
- Prolijidad en el código.
- Uso adecuado de constantes.
- Correcto diseño de la base de datos.
- Buena estructuración del código (ej: organización de carpetas, correcta separación de funcionalidades en funciones, etc).
- Coherencia en los nombres de variables, funciones, etc.
- **Uso correcto de las etiquetas semánticas de HTML.**
- Estilización del sitio.
- Prolijidad en la organización de la carpeta del proyecto.

Modalidad de entrega

La entrega se realizará de manera digital, subiendo un zip/rar con una copia del proyecto completo a donde se indique en el DV Panel, con el formato de nombre de archivo: "**apellido-nombre_final_.[zip|rar]**" (ej: gallino-santiago_final_.rar).

En caso de ser grupal, el formato es:
"apellido1-nombre1_apellido2-nombre2_final_.[zip|rar]" (ej:
gallino-santiago_noto-federico_final_.rar).

Ese archivo debe contener:

- Carpeta "sitio" con el sitio completo.
- Carpeta "db" con el archivo SQL para importar.
- Carpeta "der" con el DER de la base de datos.
- Archivo **datos.txt** con todos los datos del estudiante:

Carrera, materia, cuatrimestre, año, turno, comisión, apellido y nombre, docente, carácter de entrega (2do parcial). Además del email y password del usuario administrador para ingresar al panel.

El incumplimiento de cualquiera de los requisitos indicados en la modalidad de entrega incurrirá en una penalización de al menos 1 (un) punto menos.

El final puede incluir preguntas a los alumnos, incluyendo:

- Teóricas sobre la materia.
- Teóricas sobre materias previas relacionadas (ej: HTML).
- Sobre el trabajo, su código, las decisiones de desarrollo, etc.

Si un trabajo se detecta que es una copia de otro, automáticamente recibirá una calificación de 1 (uno). Si además en la misma mesa se presentan copiator y copiado, ambos recibirán esa calificación.