

Documentación KinCaps (Sprint #3)

Descripción:

Se ha organizado por áreas para facilitar la navegación, desde las modificaciones en el backend y la base de datos, hasta el desarrollo del Frontend, incluyendo estilos y estructura.

Alcance y Tecnologías:

- **Arquitectura y Backend:** Las modificaciones abarcan el backend en Java, incluyendo CookieAuth, LoginServlet, y CookieAuthDAO.
- **Base de Datos:** Se realizaron cambios en SQL y migraciones, afectando la tabla cookieAuth y fragmentos de la tabla factura.
- **Frontend:** Los cambios impactan las plantillas JSP del catálogo, las páginas de políticas y términos, los estilos generales en CSS y la lógica de filtros en JavaScript.

Modificaciones en el Backend:

LoginServlet:

- **Lectura del token:** Antes, se buscaba un usuario por id y tipo. Ahora, la lógica intenta obtener el usuario directamente desde el token.
- **Creación del token:** Se genera un nuevo validador y se crea un objeto CookieAuth con referencias directas al cliente o empleado. El servlet utiliza

CookieAuthDAO:

- **Gestión de Sesión:** Si el token es válido, se establece la sesión con el objeto de usuario y se redirige según si es una instancia de Cliente o Empleado.

Modelo CookieAuth:

- **Migración Conceptual:** El modelo evolucionó desde almacenar identificadores primitivos hacia relaciones ManyToOne con las entidades Cliente y Empleado.
- **Propiedades:** Se añadieron propiedades que apuntan a las entidades Cliente y Empleado, y se actualizaron los constructores y métodos getter/setter.

Estado en Entidades:

Se modificó Cliente y Carrito para agregar un estado, como activo/inactivo, y un estado pagado para los carritos.

Modificaciones en la Base de Datos:

Tabla cookieAuth:

- Se introdujeron columnas para referenciar directamente a idCliente e idEmpleado.

- Se añadió una restricción
- CHECK para forzar que solo una de las dos referencias pueda ser no nula a la vez.

Tabla factura:

- Se modificó la representación del campo

Desarrollo del Frontend:

1. JavaScript

- **Filtro del Catálogo:**
 - Inicialización: El script se ejecuta después de que el DOM esté completamente cargado.
 - Control del Slider de Precio: Lee los valores de las dos sliders para asegurar que siempre haya un mínimo y un máximo. Actualiza las etiquetas de texto para mostrar "Min" y "Max" y colorea la pista de la slider para representar el rango seleccionado.
 - Filtrado de Productos: Al detectar un evento input, el script obtiene los valores de precio, marcas y categorías seleccionadas. Luego, itera sobre los elementos .product-item y alterna su visibilidad comparando sus data-attributes.
 - Precondición Técnica: El script requiere que cada tarjeta de producto en el HTML contenga los data-attributes mencionados para funcionar.
- **Botón de Cierre de Sesión:**
 - Se añadió un script que, al pasar el cursor sobre el botón de bienvenida, cambia el texto para preguntar al usuario si desea cerrar la sesión.
 - El script se carga con el atributo defer para optimizar el rendimiento.

2. CSS

- **Principios Generales:**
 - Paleta de Colores: Centrada en degradados azules para la cabecera y un azul claro para textos destacados.
 - Tipografía: La fuente principal es 'Segoe UI' sobre un fondo global claro.
- **Componentes de Layout:**
 - Barra de Navegación: Se mantiene fija en la parte superior al hacer scroll. Los enlaces tienen un color azul claro y peso semi-negrita, cambiando a un tono más brillante con un subrayado en el evento :hover.
 - Sección Hero: Tiene una altura fija de 450px y un fondo con degradado vertical. El texto interior incluye una sombra para mejorar la legibilidad.

- Panel de Filtros: Panel izquierdo con fondo whitesmoke, bordes redondeados, sombra y posición sticky.
- Modal: Estilos para un modal que actúa como un panel lateral deslizable.
- **Elementos de Interfaz:**
 - Tarjetas de Producto: Tienen un borde inicialmente transparente que se vuelve azul en :hover. Se aplica una transición suave para elevar la tarjeta y añadir una sombra. Las imágenes usan object-fit para mantener las proporciones.
 - Insignias (.insignia-tipo): Elemento decorativo rotado 45 grados en la esquina superior derecha de las tarjetas. Se definen clases de color específicas como .tipo-f1, .tipo-toyota, etc.
 - Botones: Los botones tienen efectos de transición y sombra en :hover para mejorar la retroalimentación visual.

3. JSP:

- **Estructura General:** Las páginas utilizan un layout de dos columnas para el catálogo con una barra de navegación fija y un pie de página.
- **Navegación:** Las rutas de los enlaces se actualizaron de URLs estáticas a rutas más limpias, utilizando para mayor portabilidad.
- **Contenido Dinámico:**
 - Se utiliza JSTL para mostrar dinámicamente el historial de pedidos. Si no hay pedidos, muestra un mensaje alternativo con un enlace al catálogo.
 - Las fechas se formatean en español usando <fmt:formatDate>.
 - Los estados de los pedidos se muestran con insignias de colores que cambian según el valor.
- **Páginas Adicionales:** Se crearon páginas para "Política de Privacidad" y "Términos y Condiciones" con contenido estructurado en acordeones.

5. Observaciones Generales y Puntos de Mejora

- **Código Duplicado:** Se detectó código HTML repetido en la generación de tarjetas de productos y en la sección "Hero".
- **Inconsistencias en HTML:** Se observaron etiquetas <footer> duplicadas y anidamientos incompletos en algunos fragmentos.
- **Errores Lógicos:** Se encontraron condiciones JSTL duplicadas, lo que podría afectar el comportamiento en la visualización de estados.
- **Mezcla de Tecnologías:** Se identificó el uso mixto de scriptlets de Java con JSTL en el manejo de fechas, lo cual podría unificarse.