
PROYECTO INTERMODULAR

**IVÁN AYUSO OLIVERA
ENRIQUE AZORÍN CASTELLANO**

ARTES DORADAS



ÍNDICE

1) INTRODUCCIÓN.....	2
2) BASES DE DATOS.....	3
a) <i>Modelo Entidad/Relación.....</i>	3
b) <i>Modelo relacional.....</i>	8
- <i>Tablas.....</i>	8
- <i>Pérdidas semánticas.....</i>	10
3) PROGRAMACIÓN.....	11
a) <i>Diagramas UML.....</i>	11
- <i>DTO.....</i>	11
- <i>DAO.....</i>	14
- <i>Controlador.....</i>	16
b) <i>Explicación de la funcionalidad de la aplicación con capturas de pantalla.....</i>	19
4) LENGUAJE DE MARCAS.....	39
a) <i>Mockup.....</i>	39
b) <i>Diagrama de flujo de pantallas.....</i>	47
5) CONCLUSIONES Y DIFICULTADES ENCONTRADAS.....	48
6) POSIBLES MEJORAS O AMPLIACIONES.....	50
7) BIBLIOGRAFÍA.....	52

MEMORIA

1) INTRODUCCIÓN

Nuestro proyecto interdisciplinar consiste en una tienda online de venta de libros y discos de vinilo. En vez de hacer una librería o una disquería, decidimos juntar las dos en una sola y ofrecer como nuestro producto ambos: libros y vinilos.

Estos serán nuestros principales productos, que a su vez serán su propia categoría, es decir, cuando tu vayas a seleccionar una categoría elegirás entre libros o vinilos. Después, dentro de dichas categorías, estarán sus subcategorías que en este caso serán los géneros. Obviamente, los libros y los vinilos tendrán géneros diferentes.

- Para los libros serán: fantasía, romance, ciencia ficción, terror, etc.
- Mientras que para los vinilos: pop, rock, electrónica, hip hop, country, jazz, etc.

Estos serán los criterios de búsqueda, elegir entre libro o vinilo y luego navegar sobre los diferentes géneros de sus respectivos.

Siguiendo con más características de estos productos, cada uno de ellos podrá tener una serie de opiniones escritas por los usuarios: si les han gustado o no, si lo recomiendan... U opiniones sobre el servicio de la tienda: si los productos han llegado en buenas condiciones, comparaciones calidad-precio... En resumen, cada cliente podrá poner por escrito sus opiniones tanto del producto como de nuestro servicio.

Otra característica más será la capacidad de guardar estos productos en listas de deseos o bien seleccionarlos en una cesta de la compra. Tanto la una como la otra serán creadas por los propios usuarios.

Respecto a nuestros usuarios habrá dos tipos: administrador y cliente.

El usuario de tipo cliente será el que tenga la capacidad de realizar todas las acciones que hemos mencionado anteriormente. Navegar por los distintos productos, escribir una opinión y guardarlos en listas de deseos o llevarlos a la cesta de la compra para así realizar un pedido.

Para que este pedido sea posible, necesitamos una serie de datos: la cesta con los productos y una dirección (a la que llevar el producto). Como dato opcional, un cliente podrá vincular una tarjeta (con la que se realizará el pago del pedido).

Finalmente, se generará una factura cada vez que se realice un pedido. Esta factura se enviará también al cliente (a la misma dirección del pedido o a otra distinta) y podrá pagarla una vez le llegue en el caso de no haberla pagado con tarjeta.

2) BASES DE DATOS

a) Modelo Entidad/Relación

Nuestro proyecto consiste en realizar una tienda online de venta de libros y discos de música; por lo tanto nuestra base de datos deberá tener las siguientes *entidades*: PRODUCTOS (LIBRO; DISCO DE MÚSICA), OPINIÓN, CATEGORÍA, LISTA DE DESEOS, PEDIDO, USUARIO (ADMINISTRADOR; CLIENTE), DIRECCIÓN, FACTURA Y TARJETA.

A continuación se verán los *atributos* de cada *entidad*, y explicaremos las relaciones que hay entre cada una de ellas.

Los *atributos* de las *entidades* serán los siguientes:

- Respecto a los productos queremos saber lo siguiente: tendrán un código de producto (que los identificará), un nombre, una descripción (que tendrá sus características), su precio, la unidad de medida de dicho producto (UNIDADES), su IVA, la cantidad de la que disponemos (Stock), la cantidad mínima que deberá haber de cada producto, una foto.

Todos estos datos serán obligatorios, a parte de estos, habrá una serie de datos opcionales que serán los siguientes:

- Su fecha de creación.
- La fecha de la última vez que ha sido modificado.

Estos productos serán de dos tipos: libros y discos de música. Esto será una *generalización*.

De los libros queremos saber: Género, autor, número de páginas, ISBN, editorial, formato.

De los discos de música: Género, artista, canciones, sello, discográfica, ASIN.

- Cada producto pertenece a una categoría, de la que se querrá saber:
 - Código (con la que se identifica).
 - Nombre (Libro o vinilo).

Las categorías tendrán subcategorías (con nombre propio).

- Código (su identificador).
- Nombre (el género).

- Un producto podrá tener opiniones. Los atributos de una opinión son:
 - Número (identificador).
 - Mensaje.
- Las listas de deseos, se identificarán con un nombre.

- Respecto a los pedidos queremos saber: su código de pedido (su identificador), la fecha en la que se realiza, el precio total del pedido, la dirección del pedido.
- Los pedidos generarán una factura. De la factura necesitamos: su código de factura (identificador), la fecha en la que se ha realizado, una dirección de facturación. (podrá ser distinta de la del pedido).
- Para solucionar que hay dos posibilidades para una dirección (que sea de pedido o de factura), la convertiremos en una entidad. De la dirección queremos saber:
 - Número (será el identificador).
 - Tipo (pedido o factura).
 - CP.
 - Localidad.
 - Provincia.
 - Dirección.

Esta dirección estará asociada a un usuario.

- Usuarios: se identificarán con un código, vincularán un correo electrónico (ha de ser único), necesitan una contraseña, queremos saber su nombre completo y la última vez que se conectaron (última conexión).

A parte, habrá una serie de datos opcionales:

- Teléfono.
- Fecha de nacimiento.
- Foto.

Un usuario podrá ser de dos tipos: cliente o administrador. Esto será una *generalización*.

- Un usuario también podrá vincular una tarjeta, de la que se querrá saber su número.

Las relaciones de las diferentes *entidades* serán las siguientes:

- Un *producto* podrá o no tener una *opinión*, y podrá tener varias *opiniones*. A su vez, una *opinión* habrá de estar si o si asociada a un *producto*, y cada *opinión* será única para cada *producto*. Por lo tanto, tendrá una *cardinalidad 1:N*. Como un *producto* puede tener varias *opiniones* necesitamos saber a qué *producto* se está refiriendo para poder identificarla ya que la *opinión* 02 no necesariamente tiene que ir relacionada con el *producto* 02, por lo tanto será una *restricción de ID*.

OPINIÓN	PRODUCTO
01	01
01	02
02	01
02	02

- Estas *opiniones* son escritas por los *usuarios*, su relación seguirá la misma lógica que con los productos.
- Un *producto* pertenece a una *categoría* como mínimo, no puede haber *productos* sin *categoría* (Ya que hemos mencionado que las *categorías* serán los propios *productos* (Libros y vinilos). Por lo tanto, un *producto* tendrá a su vez una *categoría* como máximo (*cardinalidad 1:1*). Esto quiere decir que no puede haber una *categoría* sin *producto* o un *producto* sin *categoría* (*restricción de existencia*)).
- Respecto a las *subcategorías*, una *categoría* tendrá sí o sí una *subcategoría*, ya que tanto los libros como los vinilos tienen como mínimo un género asociado (el género serán las *subcategorías*). Habrá la misma cantidad de *subcategorías* que de géneros dispongamos de nuestros diferentes *productos*. Una *subcategoría* es exclusiva de una *categoría* (no pueden haber libros de género rock al igual que no hay vinilos de género de fantasía), y una *subcategoría* no existirá sin categoría asociada, por lo tanto habrá una restricción de existencia y tendrá una *cardinalidad 1:N*.
- A una *lista de deseos* se le podrán añadir como mínimo un *producto* y como máximo muchos. Un *producto* no tiene porque estar añadido a una *lista de deseos*, pero, podrá ser añadido a muchas. Con esto tenemos una *cardinalidad N:N*.
- Dicha *lista de deseos* será creada por un *cliente* y será única para cada uno, y un *cliente* podrá crear muchas, pero no necesariamente habrá de crear una. *Cardinalidad 1:N*. En resumen, una lista de deseos no existirá si no la crea un *cliente* y será una *restricción de ID*, ya que al estar identificadas con un nombre, necesitamos saber que *cliente* a creado dicha *cesta*:

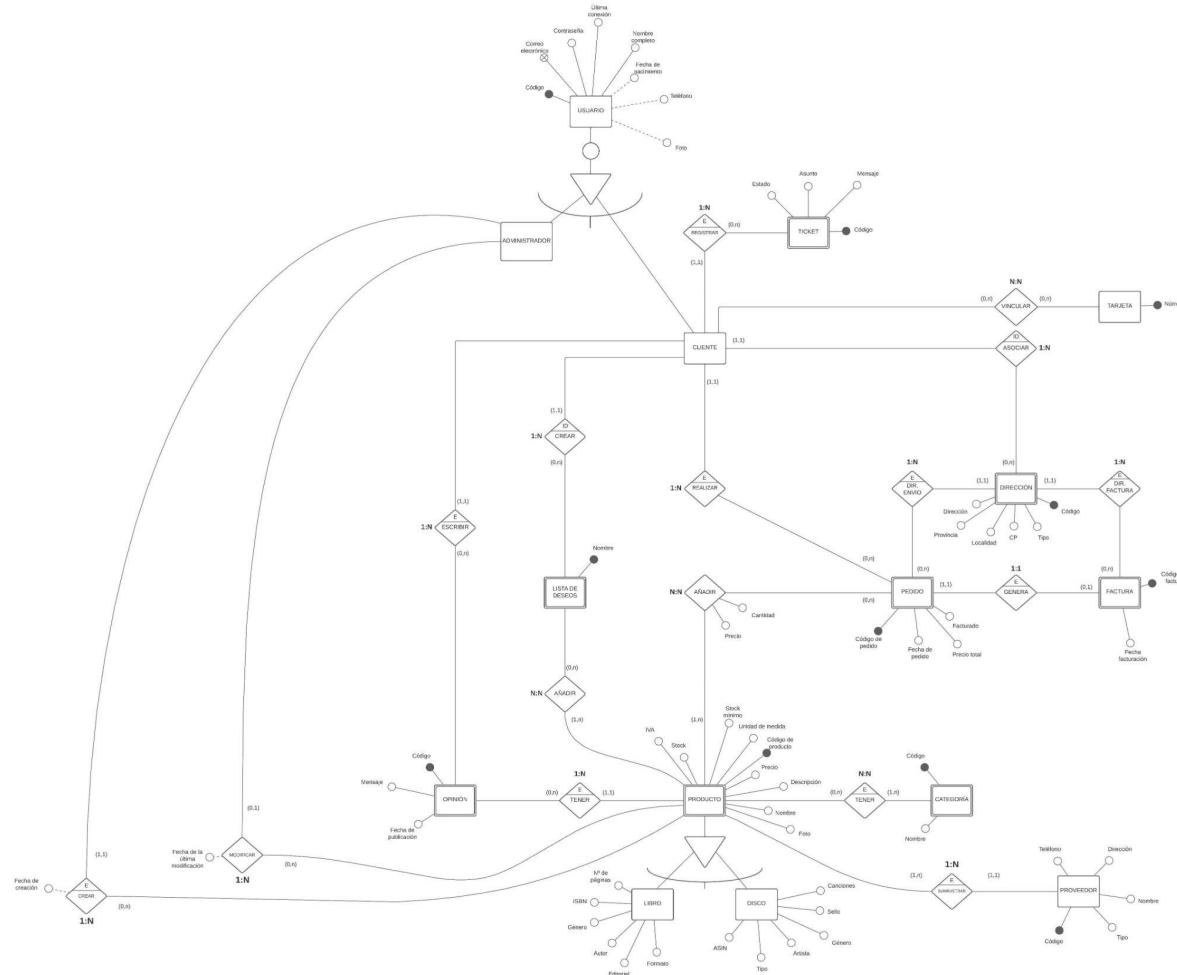
LISTA DE DESEOS	CLIENTE
Lista de reyes	01
Lista de reyes	02
Lista de cumpleaños	01
Lista de cumpleaños	02

- Un *pedido* existirá solamente si se ha realizado una *cesta* (*restricción de existencia*), esté será único de cada *cesta*. *Cardinalidad 1:1*. Una *cesta* no necesariamente tiene que realizar un *pedido* (tú puedes crear una *cesta*, pero aun así no realizar el *pedido*). Además, un *pedido* realizado generará una *factura*, que no existirá si no se ha realizado dicho *pedido* (*restricción de existencia*). Cada *factura* será única de cada *pedido* y viceversa. *Cardinalidad 1:1*.
- Un *cliente* podrá tener vinculadas varias *tarjetas* (no necesariamente tiene que tener una) y viceversa. Esto será una *cardinalidad N:N*. Dicho *cliente* también podrá tener o no asociadas varias *direcciones* y una *dirección* debe pertenecer a un *cliente* y será únicamente de este. *Cardinalidad 1:N*. Como un *cliente* puede tener varias *direcciones* necesitamos saber a qué *cliente* están asociadas para poder identificarlas ya que la *dirección* 02 no necesariamente tiene que ir relacionada con el *cliente* 02, por lo tanto será una *restricción de ID*.

DIRECCIÓN	CLIENTE
01	01
01	02
02	01
02	02
03	02

- Una *dirección* irá sí o sí asociada a una *factura* o a un *pedido* (podrán ser *direcciones* diferentes) y será única de cada una. Como hemos dicho que pueden ser diferentes, una *dirección* no tiene porque estar como dirección de pedido, si no que será la dirección de facturación y otra diferente será la de pedido. Una *dirección* podrá estar en muchos *pedidos* y *facturas*, *cardinalidad 1:N*. Además, tanto *factura* como *pedido* no existirán si no hay *dirección*, *restricción de existencia*.

Enlace: [Lucidchart](#)



b) Modelo relacional

Todos los atributos son considerados “no nulos” excepto los opcionales. Estos atributosopcionales estarán marcados con un “*”.

- Tablas

- USUARIOS (codigo, correo_electronico, password, ultimaConexion, nombre, fecha_nacimiento*, telefono*, foto*)
 - PK: codigo
 - UK: correo_electronico
- ADMINISTRADORES (codigo)
 - PK: codigo
 - FK: codigo → USUARIO.codigo
- CLIENTES (codigo)
 - PK: codigo
 - FK: codigo → USUARIO.codigo
- TICKETS (codigo, mensaje, asunto, estado, cliente)
 - PK: codigo
 - FK: cliente → CLIENTE.codigo
 - NN (cliente)
- TARJETAS (numero)
 - PK: numero
- TARJETAS_USUARIOS (cliente, tarjeta)
 - PK: cliente, tarjeta
 - FK: cliente → CLIENTES.codigo
 - FK: tarjeta → TARJETAS.numero
- DIRECCIONES (cliente, codigo, tipo, cp, localidad, provincia, direccion_completa)
 - PK: cliente, codigo
 - FK: cliente → CLIENTES.codigo
- PEDIDOS (codigo, fecha, precio_total, facturado, cliente, cliente_envio, direccion_envio)
 - PK: codigo
 - FK: cliente → CLIENTES.codigo
 - NN (cliente)
 - FK: cliente_envio, direccion_envio → DIRECCIONES.cliente, codigo
 - NN (cliente_envio, direccion_envio)
- FACTURAS (codigo, fecha, cliente, direccion_factura, pedido)
 - PK: codigo
 - UK: pedido
 - FK: cliente, direccion_factura → DIRECCIONES.cliente, codigo
 - NN (cliente, direccion_factura)
 - FK: pedido → PEDIDOS.codigo
 - NN (pedido)
- LISTAS_DESEOS (cliente, nombre)
 - PK: cliente, nombre
 - FK: cliente → CLIENTES.codigo

- OPINIONES (codigo, mensaje, fecha_publicacion, cliente, producto)
 - PK: codigo
 - FK: cliente → CLIENTES.codigo
 - NN (cliente)
 - FK: producto → PRODUCTOS.codigo
 - NN (producto)
- PROVEEDORES (codigo, telefono, direccion, nombre, tipo)
 - PK: codigo
- PRODUCTOS (codigo, foto, nombre, descripcion, precio, unidad_medida, stock_minimo, stock, iva, proveedor, modificador*, creador, fecha_creacion, fecha_ultima_modificacion*)
 - PK: codigo
 - FK: proveedor → PROVEEDORES.codigo
 - NN (proveedor)
 - FK: modificador → ADMINISTRADORES.codigo
 - FK: creador → ADMINISTRADORES.codigo
 - NN (creador)
- LIBROS (codigo, formato, editorial, autor, genero, isbn, numero_paginas)
 - PK: codigo
 - FK: codigo → PRODUCTOS.codigo
- DISCOS (codigo, canciones, sello, genero, artista, tipo, asin)
 - PK: codigo
 - FK: codigo → PRODUCTOS.codigo
- CATEGORIAS (codigo, nombre)
 - PK: codigo
- CATEGORIAS_PRODUCTOS (categoria, producto)
 - PK: categoria, producto
 - FK: categoria → CATEGORIAS.codigo
 - FK: producto → PRODUCTOS.codigo
- PEDIDOS_PRODUCTOS (producto, pedido, cantidad, precio)
 - PK: producto, pedido
 - FK: producto → PRODUCTOS.codigo
 - FK: pedido → PEDIDOS.codigo
- LISTAS_DESEOS_PRODUCTOS (producto, nombre_lista, cliente)
 - PK: producto, nombre_lista, cliente
 - FK: producto → PRODUCTOS.codigo
 - FK: nombre_lista, cliente → LISTAS_DESEOS.cliente, nombre

- Pérdidas semánticas

- Generalización → solamente hay pérdida si una generalización es total o exclusiva.

La generalización de USUARIO es tanto total como exclusiva. Se pierden tanto la totalidad como la exclusividad puesto que no podemos plasmarlas en el modelo relacional. La generalización de PRODUCTO es exclusiva. Se pierde la exclusividad ya que no se puede plasmar en el modelo relacional.
- (1,n) → Se pierde el uno, no se puede representar. Ya que en el relacional sólo podemos captar la existencia mínima de ese producto en relaciones con restricción de existencia.
 - Se pierde el 1 del (1,n) de PRODUCTOS-PROVEEDORES.
 - Se pierde el 1 del (1,n) de PRODUCTOS-LISTAS_DESEOS.
 - Se pierde el 1 del (1,n) de PRODUCTOS-PEDIDOS.
 - Se pierde el 1 del (1,n) de CLIENTES-PEDIDOS.
- 1:N o 1:1 → Si hay atributo en la relación, no se puede saber su origen. En las relaciones 1:N ADMINISTRADOR-MODIFICAR-PRODUCTO y ADMINISTRADOR-CREAR-PRODUCTO, se pierde el origen de los atributos “Fecha de creación” y “Fecha de última modificación” porque no hacemos una tabla de la relación.

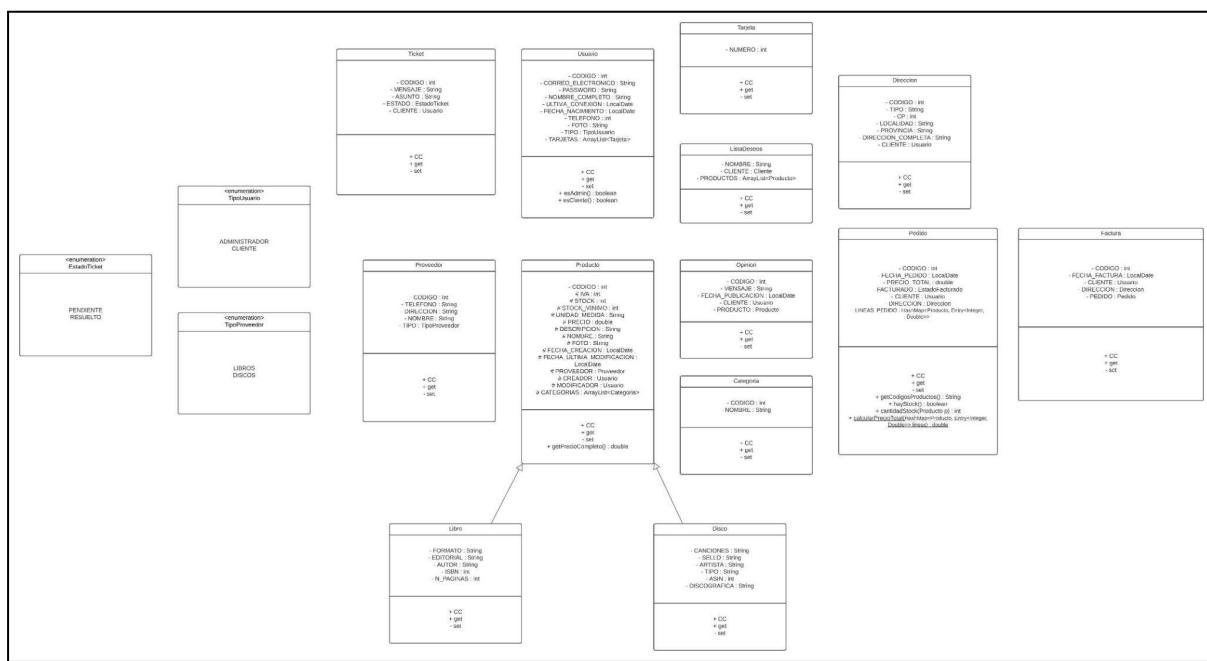
3) PROGRAMACIÓN

a) Diagramas UML

- DTO

Lucidchart:

https://lucid.app/lucidchart/a5842b39-deed-4355-b6a8-27899b03e32f/edit?viewport_loc=260%2C616%2C6656%2C3096%2C3v2yRcT1XMKM&invitationId=inv_133b6726-de55-4de9-8f80-93baa7fd3e37



Las clases del DTO son aquellas que implementarán la estructura de las diferentes tablas de la base de datos, para poder traspasar los datos que contienen las diferentes tablas de la base de datos a objetos con los que podemos dar uso a las diferentes funcionalidades del programa.

Las diferentes clases del DTO son:

Todas las clases tendrán constructores completos y getter de todos sus atributos, además implementarán los métodos “equals” y “hashCode” para poder comprobar si dos objetos son iguales.

Además, también implementarán el método “toString” por si se necesitaría en algún momento poder imprimir por pantalla los valores de todos los atributos (También puede ser usado para realizar comprobaciones”.

Otro factor importante es que los atributos de todas las clases serán constantes, para evitar así la modificación de datos y tener que estar construyendo nuevos objetos para realizar cualquier modificación.

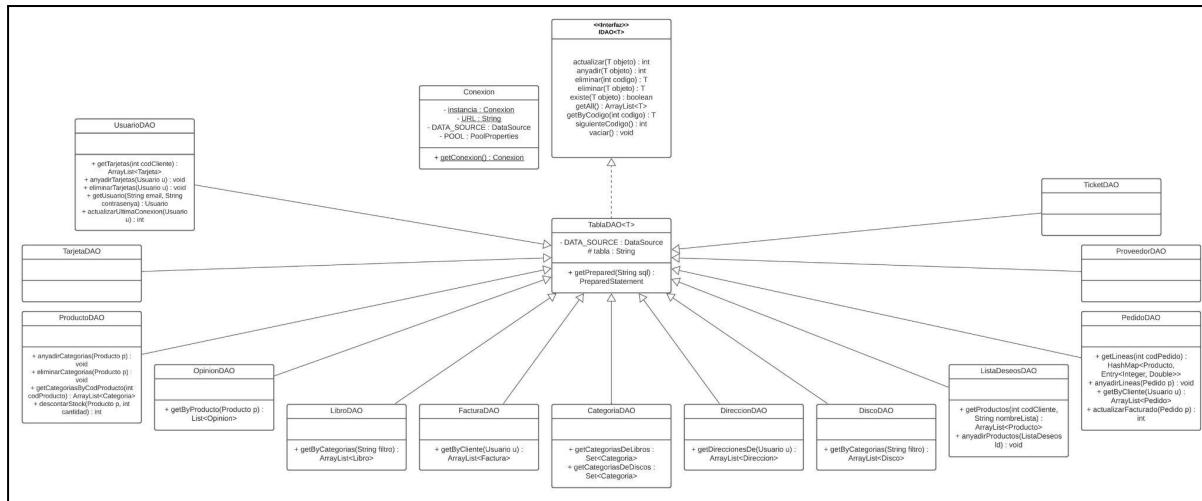
- Categoría
 - Esta clase tendrá los atributos “tipo” y “código”, que serán los mismos que los de la tabla categorías.
 - La clase de Categoria implementará Comparable ya que necesitaremos el método “compareTo”, porque realizamos operaciones con conjuntos de categorías (Set).
- Dirección
 - Sus atributos serán los mismos que los de la tabla direcciones; código, tipo, cp, localidad, provincia y dirección completa y cliente.
- Disco
 - Es una de las dos clases que hereda de productos, obteniendo así todos sus atributos y métodos.
 - Por necesidades a la hora de realizar operaciones con esta clase, dispondrá de un constructor al que le entrará un producto y los datos propios de los discos.
 - Los atributos de los que dispondrá la clase son: canciones, sello, artista, tipo, asin y discográfica.
- EstadoTicket
 - Esta clase es un ENUM y implementará el check que tenemos en bases de datos que impide que el estado de un ticket sea otro que “RESUELTO” o “PENDIENTE”.
- Factura
 - Tendrá la estructura de la tabla de facturas, sus atributos serán: codigo, fecha de factura, cliente, dirección y pedido.
- Libro
 - Libro es la otra clase que hereda de producto y su estructura será igual que la de disco.
 - Los atributos que tendrá son: formato, editorial, autor, isbn y número de páginas.
- ListaDeseos
 - Es una clase que combina dos tablas “listas_deseos” y “listas_deseos_productos”.
 - Para implementar la relación N:N, esta clase tendrá como atributo un ArrayList de productos, además de los propios de la tabla listas: nombre y cliente.
- Opinion
 - Tendrá como atributos los mismos que los de la tabla opiniones (codigo, mensaje, fecha de publicación, cliente y producto).

- Pedido
 - Tendrá la estructura de la tabla de pedidos.
 - Sus atributos serán: código, fecha de pedido, precio total, facturado, cliente, dirección.
 - Además, tendrá un atributo que será un HashMap, que guardará un producto con su cantidad y precio, implementando la tabla “pedidos_productos”.
 - Pedidos tendrá métodos adicionales:
 - getCodeProducts: este método recorre el HashMap y obtendrá un String con todos los códigos de los productos que lo componen separados con comas.
 - hayStock: recorrerá el HashMap y comprobará que hay suficiente stock para cada producto que lo componga.
 - cantidadStock: obtendrá la diferencia entre la cantidad que se pide de un producto en un pedido y la cantidad de stock de dicho producto.
 - calcularPrecioTotal: calculará el precio total de la línea de productos.
- Producto
 - Es la clase padre de discos y libros y la que implementa la estructura de la tabla pedidos, teniendo los atributos: código, iva, stock, stock mínimo, unidad de medida, precio, nombre, descripción, foto, fecha de creación, fecha de modificación, proveedor, creador, modificador.
 - Además, para implementar la relación N:N, es decir, la tabla “categorias_productos”, dispondrá de un atributo ArrayList de categorías.
 - Producto tendrá un método adicional: getPrecioCompleto, que calculará el precio que tendrá el producto añadiendo el IVA.
- Proveedor
 - Proveedor tendrá los datos de la tabla proveedores, sus atributos son: código, nombre, teléfono, dirección y tipo.
- Tarjeta
 - Tarjeta tendrá como atributo el número, igual que la tabla “tarjetas”.
- Ticket
 - Ticket es la clase que obtendrá los datos de la tabla “tickets”: código, mensaje, asunto, estado y cliente.
- TipoProveedor
 - Otro de los ENUMS, sus datos son LIBROS y DISCOS, para implementar el check que tenemos en la base de datos de que un proveedor sólo puede ser uno de esos dos tipos.
- TipoUsuario
 - ENUM. Será ADMINISTRADOR o CLIENTE, otro check de la base de datos, validando que el usuario solo pueda ser o bien cliente o bien administrador.
- Usuario
 - Última clase del DTO. Su estructura es la misma que la de la tabla “usuarios”. Con atributos: código, correo electrónico, contraseña, nombre, fecha de nacimiento, teléfono, foto, tipo y última conexión.
 - Tendrá otro atributo, un ArrayList de tarjetas para implementar la tabla “tarjetas_usuarios”.
 - Dispondrá de dos métodos adicionales:
 - esAdmin: devolverá “true” si el usuario se trata de un administrador.
 - esCliente: devolverá “true” si el usuario se trata de un cliente.

- DAO

Lucidchart:

https://lucid.app/lucidchart/a5842b39-deed-4355-b6a8-27899b03e32f/edit?viewport_loc=-1213%2C32%2C2994%2C1495%2Cs.o3chHg9pZV&invitationId=inv_133b6726-de55-4de9-8f80-93baa7fd3e37



Las clases del DAO son las que se relacionan de forma aún más directa con la base de datos. Tienen como objetivo, obtener los diferentes registros de las tablas que componen nuestra base de datos, o bien traspasar los datos de las clases de Java para realizar operaciones en la base de datos como inserciones, actualizaciones, borrados, etc. Las clases que componen el DAO son una por cada una de las tablas que haya en la base de datos: CategoriaDAO, DireccionDAO, DiscoDAO, FacturaDAO, LibroDAO, ListaDeseosDAO, OpinionDAO, PedidoDAO, ProductoDAO, ProveedorDAO, TarjetaDAO, TicketDAO y UsuarioDAO. Además de tres clases adicionales: Conexion, IDAO y TableDAO.

Todas las clases del DAO tendrán un atributo de tipo String que será “tabla”, que contendrá el nombre de la tabla según se relaciona con la clase. Ej: CategoriaDAO será “ARTESDORADAS_categorias”.

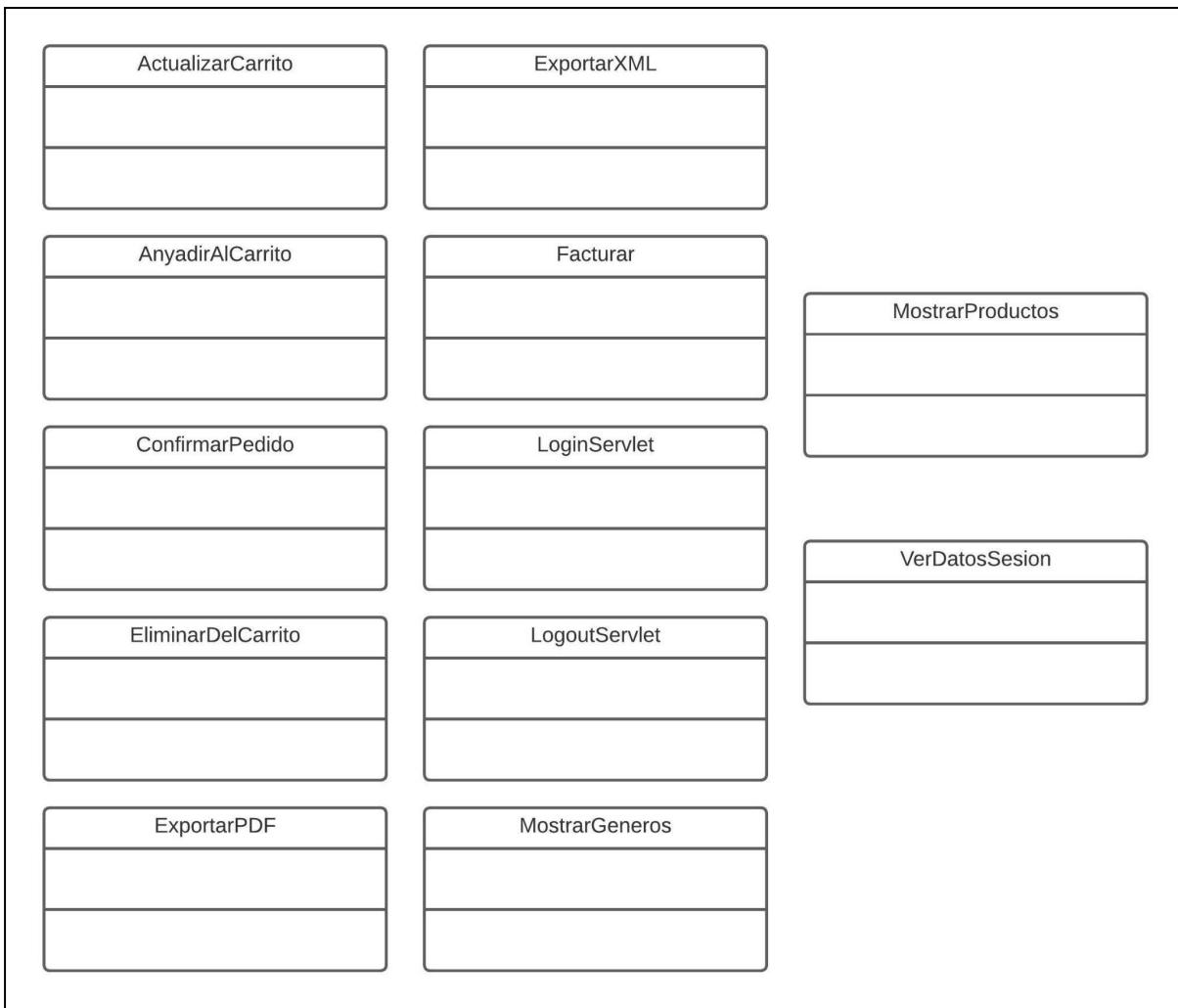
- Conexión
 - Su función es la de conectarse a la base de datos para que las demás clases puedan realizar operaciones con esta.

- **IDAO**
 - IDAO es una interfaz que tendrá métodos abstractos comunes para todas las demás clases DAO.
 - actualizar: se encargará de realizar un UPDATE en la base de datos.
 - anyadir: se encargará de realizar un INSERT en la base de datos.
 - eliminar: realizará un DELETE en la base de datos.
 - existe: su función es comprobar que un dato existe en nuestra base de datos.
 - getAll: obtendrá todos los registros de una tabla.
 - getByCodigo: obtendrá un registro según el código que se le proporcionará.
 - siguienteCodigo: calculará cuál será el siguiente código que se deberá INSERTAR en una tabla.
 - vaciar: vaciará por completo todos los registros de una tabla.
- **TablaDAO**
 - Mediante el método “getPrepared” se encargará de comenzar la conexión con la base de datos.
 - Implementará algunos métodos que no dependen específicamente de una tabla en concreto, como por ejemplo eliminar, existe, siguienteCodigo o vaciar.
- **CategoriaDAO**
 - Tendrá dos métodos adicionales: getCategoriasDeLibros y getCategoriasDeDiscos.
 - Estos dos métodos se encargan de obtener todas las categorías que pertenezcan a libros o a discos.
- **DireccionDAO**
 - Se le implementará un método adicional en el que le entrará un cliente y se obtendrán todas sus direcciones: getDireccionesDe.
- **DiscoDAO, LibroDAO**
 - Compartirán un método que se llamará “getByCategorias”, que devolverá todos los registros que tengan algunas de las categorías proporcionada.
- **FacturaDAO**
 - Tendrá un método que obtendrá todas las facturas de un cliente en concreto: getByCliente.
- **OpinionDAO**
 - El método adicional es getByProducto, que obtendrá todas las opiniones generadas en un producto específico.
- **PedidoDAO**
 - Dispondrá de métodos adicionales para poner insertar, actualizar y borrar las líneas. Además, de un método getByCliente.
 - Otro método adicional será actualizarFacturado que actualizará el estado del pedido en la base de datos.
- **ProductoDAO**
 - Un método adicional descontarStock que realizará un UPDATE para poder modificar el stock de un producto en la base de datos.
- **UsuarioDAO**
 - Tendrá métodos para poder insertar, actualizar y eliminar tarjetas.
 - Otro para actualizar la fecha de última conexión, actualizarFechaUltimaConexion
 - Finalmente, se le añade un método para obtener un usuario según un correo y una contraseña.

- Controlador

Lucidchart:

https://lucid.app/lucidchart/a5842b39-deed-4355-b6a8-27899b03e32f/edit?viewport_loc=-1309%2C-759%2C5447%2C2720%2ChFp3DR72wZuj&invitationId=inv_133b6726-de55-4de9-8f80-93baa7fd3e37



- ActualizarCarrito

El servlet intentará actualizar el carrito según los nuevos parámetros que ha modificado el usuario.

Para comenzar, verificará si hay un usuario en sesión, de no ser el caso redirigirá un mensaje de error a la vista de productos.

En caso de que sí que haya un usuario en sesión, comprobará si el carrito está vacío, en el caso de que lo esté, no realizará ninguna acción y redirigirá al usuario al carrito.

Si hay un carrito creado, y tiene algún producto, actualizará la nueva cantidad seleccionada por el usuario para ese producto, no sin antes verificar que la nueva cantidad no es negativa.

De no tratarse de una cantidad negativa, el servlet actualizará el carrito con los nuevos valores.

- **AnyadirAlCarrito**

El servlet se encargará de procesar la solicitud de un usuario para añadir productos a su carrito de la compra.

De primeras, comprobará que haya algún usuario en sesión, de no ser este el caso, redirigirá al usuario a la página de productos mostrándole un mensaje de error.

En el caso de que si haya un usuario en sesión, comprobará que haya o no un carrito creado con anterioridad, en el caso de que no haya un carrito ya creado, se iniciará uno nuevo al que se le añadirá la nueva línea solicitada por el usuario, en el caso contrario, que ya haya un carrito creado, simplemente añadirá la línea al carrito actual.

Una vez realizadas las operaciones, se redirigirá al usuario a la página del carrito con este actualizado, mostrando los nuevos productos añadidos.

- **ConfirmarPedido**

Al igual que los servlets anteriores, comprobará que haya un usuario en sesión, y de no haberlo, mostrará un mensaje de error.

En el caso de que si que haya un usuario logueado, el servlet comprobará si hay stock suficiente de cada producto para poder efectuar el pedido. En el caso de que se detecte que en alguna de las líneas, la cantidad deseada supera la del propio producto, el sistema redirigirá al usuario a la página del carrito, mostrando un error.

En caso de que haya suficiente stock para poder efectuar la compra, se descontará el stock de cada uno de los productos que haya en las líneas acorde con la cantidad que se desea para la compra. Una vez descontado el stock, se insertará en la tabla de pedidos un nuevo pedido con los datos del carrito.

- **EliminarDelCarrito**

La función de este servlet será simple, recogerá el carrito de la sesión y el código del producto a eliminar. Tras esto, eliminará del carrito el producto deseado y creará un nuevo pedido con los datos actualizados. Este pedido será guardado en el carrito, actualizado ya para no tener el producto y se redirigirá al usuario de nuevo a la vista de producto donde podrá ver el carrito sin el producto.

- **ExportarPDF y ExportarXML**

Los servlets de exportar a pdf y exportar a xml se encargarán de recoger los datos de una factura elegida por el usuario y trasladar dichos datos a un formato de pdf o xml.

- Facturar

Como todos, comprobará que haya un usuario en sesión, y no de haberlo mostrará un mensaje de error. En el caso de que si lo haya, recogerá el código del pedido el cuál el usuario ha solicitado facturar.

En el caso de que el código del cliente del pedido, no coincida con el del usuario, a no ser que se trate de un administrador, se mostrará otro mensaje de error.

Si todo funciona correctamente, se creará una nueva factura, con el pedido solicitado por el usuario, está se añadirá a nuestra base de datos de datos y se actualizará el estado del pedido de “no facturado” a “facturado”.

Una vez se hayan realizado todos los pasos, el usuario será redirigido a la vista de facturas, donde podrá ver la nueva factura añadida.

- LoginServlet

El servlet se encargará de recoger los datos del formulario de inicio de sesión y buscar en la base de datos un usuario coincidente en correo electrónico y contraseña. En el caso de no se encuentre ningún usuario registrado con esos datos, se mostrará un mensaje de error.

En el caso contrario, se actualizará la última conexión de ese usuario, para poner la fecha y hora actuales y se redirigirá al index.

- LogoutServlet

El servlet de logout se encargará de cerrar la sesión, y una vez cerrada, el usuario pasará a ser anónimo, redirigiéndolo al index.

- MostrarGeneros

Este servlet se encargará de recoger los datos del filtro de la página de productos, en concreto del apartado de categorías, según la categoría seleccionada, creará un conjunto de géneros que serán los que coincidan con esa categoría y redirigirá al usuario a la vista de productos, donde ahora en el filtro se mostrarán los géneros de la categoría seleccionada.

- MostrarProductos

Mostrar productos se encargará de recoger el conjunto de géneros que haya seleccionado en el filtro de la vista de productos, además de la categoría.

Dependiendo de la categoría seleccionada y los géneros seleccionados, el servlet se encargará de recoger de la base de datos los productos que coincidan en género con los elegidos por el usuario.

Una vez recogidos por el servlet, se redirigirá al usuario a la página de productos donde se cargarán solamente los productos recogidos tras filtrarlos por los géneros elegidos.

b) Explicación de la funcionalidad de la aplicación con capturas de pantalla



La primera vista que verá un usuario anónimo nada más entrar a la página web será la del index. En esta sección de la web el usuario pueda navegar a las diferentes secciones mediante la barra de navegación.



La barra de navegación será un elemento principal de la página web que estará presente en todas las demás vistas o secciones. Desde ella, el usuario puede moverse por la página pulsando sobre los diferentes botones, que funcionan como enlaces a las diferentes secciones. El botón de iniciar sesión será el más especial ya que te llevará al formulario de login de la página.

INICIAR SESIÓN

CORREO:

CONTRASEÑA:

Enviar

¿No tienes una cuenta? [Regístrate](#)

[Volver a inicio](#)

El formulario de login está formado por dos campos, uno para el correo electrónico y otro para la contraseña. Ambos campos deben coincidir con alguno de los datos que tengamos almacenados en la tabla de usuarios de nuestra base de datos para que el login pueda llegar a realizarse. Para ello, una vez introducidos los campos, se envía la información al servlet “LoginServlet”, que se encargará de realizar los pasos necesarios para obtener un usuario que coincida en correo electrónico y contraseña, según los valores que introdujo el usuario en el formulario de login.

En el caso de que se encuentre un usuario coincidente, se actualizará la fecha de última conexión de ese usuario y la web será redirigida de nuevo al index, donde ya se podrán apreciar cambios por el inicio de sesión.

INICIAR SESIÓN

CORREO:

Correo electrónico

CONTRASEÑA:

Contraseña

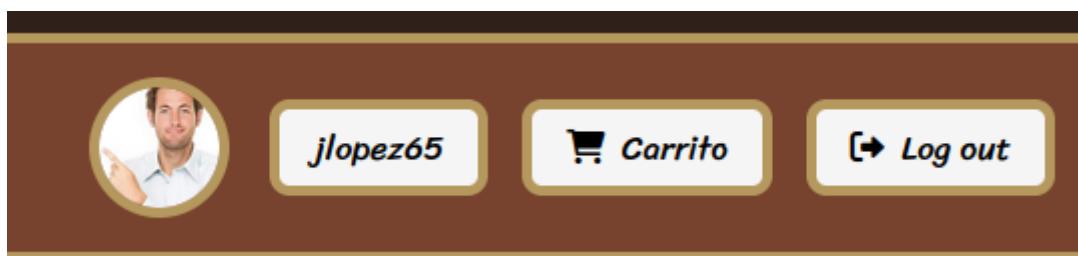
Enviar

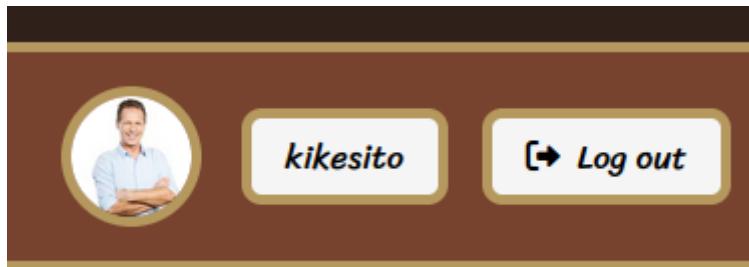
(*) Nombre de correo o contraseña inválidos. Vuelve a intentarlo

¿No tienes una cuenta? [Regístrate](#)

[Volver a inicio](#)

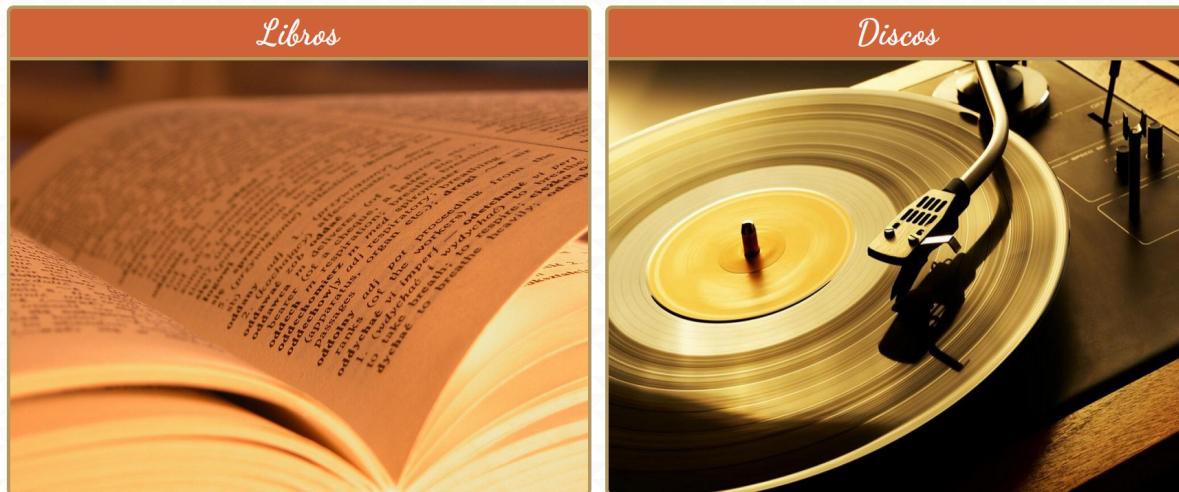
En el caso de que los datos introducidos por el usuario, es decir, el correo electrónico y la contraseña, sean erróneos, o lo que es lo mismo, no coincidan con ninguno de los registros de nuestra tabla usuarios. No se realizarán ninguna de las acciones anteriores, no se modificará la fecha de última conexión ni se redirigirá de nuevo al index; sino que se mostrará un mensaje de error en la misma página del formulario: (*) Nombre de correo o contraseña inválidos. Vuelve a intentarlo.





En el caso de que el login haya sido correcto, y se hayan ejecutado las operaciones, el usuario ahora estará logueado en la web y uno de los cambios más significativos será el del header.

El botón del login desaparecerá, dando paso a dos o tres botones nuevos (dependiendo del usuario que sea) que ocuparán su lugar: la imagen del usuario, un enlace al perfil, el carrito (solo si el usuario es un cliente) y el botón de log out.



Volviendo al índice, este también podrá cambiar con el inicio de sesión dependiendo del usuario que se haya logueado.

Si se trata de un cliente, la página principal será igual que la que ya había con los usuarios anónimos. En el centro de la pantalla, se podrán observar dos imágenes, que funcionarán como enlaces a la sección de producto de nuestra página.

Si el usuario selecciona "Libros", le llevará a la sección de productos, pero solamente se mostrarán los productos de la categoría "libro".

Los mismo para el botón de "Discos", el usuario será redirigido a la sección de productos, donde se le mostrarán los de la categoría "disco".

VER CLIENTES

VER PEDIDOS

VER PRODUCTOS

VER FACTURAS

En el caso de que se trate de un administrador, el index ya no dispondrá de esos enlaces a productos. Sino que tendrá un conjunto de cuatro botones, más simples y menos llamativos, cuya función seguirá siendo de redirección pero para secciones más significativas para un administrador.

Ver clientes llevará al administrador a una lista de todos los clientes de la web.

Ver productos redirigirá a la sección de productos.

Ver pedidos y ver facturas llevarán a páginas donde se podrán visualizar en formato tabla todos los pedidos y todas las diferentes facturas que han ido realizando todos los clientes de la tienda.



ARTES DORADAS

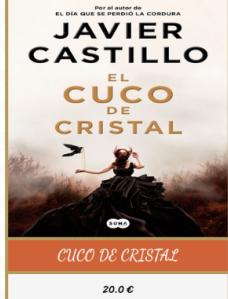
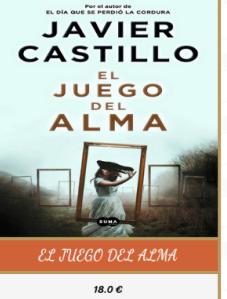
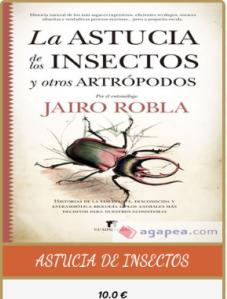
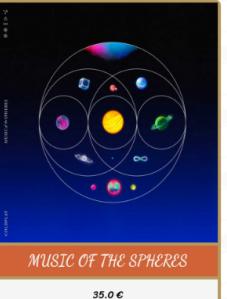
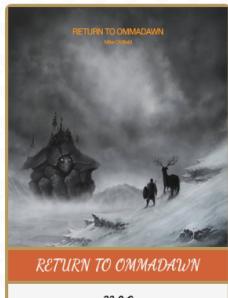
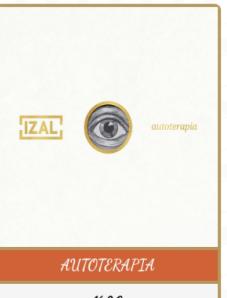
[Inicio](#) [Productos](#) [Sobre nosotros](#) [Contacto](#) [Iniciar sesión](#)

CATEGORÍAS

Libros Discos

GÉNEROS

Buscar...

 <p>JAVIER CASTILLO EL CUICO DE CRISTAL</p> <p>CUICO DE CRISTAL</p> <p>20.0 €</p>	 <p>Fernando Aramburu HIJOS DE LA FÁBULA</p> <p>colección andaluzas</p> <p>HIJOS DE LA FÁBULA</p> <p>17.0 €</p>	 <p>JAVIER CASTILLO EL JUEGO DEL ALMA</p> <p>EL JUEGO DEL ALMA</p> <p>18.0 €</p>
 <p>JAVIER CASTILLO LA CHICA DE NIEVE</p> <p>LA CHICA DE NIEVE</p> <p>15.0 €</p>	 <p>JAIRO RÖBLA LA ASTUCIA de los INSECTOS y otros ARTRÓPODOS</p> <p>ASTUCIA DE INSECTOS</p> <p>10.0 €</p>	 <p>MUSIC OF THE SPHERES</p> <p>35.0 €</p>
 <p>RETURN TO OMMANDAWN</p> <p>23.0 €</p>	 <p>SHINOVA LA BUENA SUERTE</p> <p>RESERVA YA EL NUEVO DISCO DE SHINOVA Y CONCÚBE MERCHANDISING EXCLUSIVO DE LA BANDA</p> <p>LA BUENA SUERTE</p> <p>20.0 €</p>	 <p>IZAL</p> <p>autoterapia</p> <p>AUTOTERAPIA</p> <p>16.0 €</p>
 <p>LENGLA DE PÁJAROS</p> <p>Víctor Sellés</p> <p>OBSCURA</p> <p>LA LENGUA DE LOS PÁJAROS</p> <p>21.0 €</p>		

[Términos y condiciones](#) [Política de privacidad](#)

© 2022-23, Iván Ayuso & Enrique Azorín
1º DAW | IES Poeta Paco Mollá (Petrer)

Cambiando de vista a la sección de producto, lo más principal de esta sección es que se mostrará un listado de todos los productos disponibles de nuestra tienda online.

En el caso de que se haya llegado aquí mediante el enlace de la barra de navegación o el enlace de “ver productos” de la vista principal de administrador, se mostrarán todos los productos de la base de datos sin aplicar ningún filtro. En el caso de que se entre mediante los enlaces del index a productos o discos, de la vista de usuario anónimo o cliente, se mostrarán solamente los productos que coincidan con esa categoría.

Además, una vez se haya accedido a esta páginas, se podrán filtrar aún más los diferentes productos según el filtro que se encuentra en la esquina superior izquierda.

CATEGORÍAS

Libros
 Discos

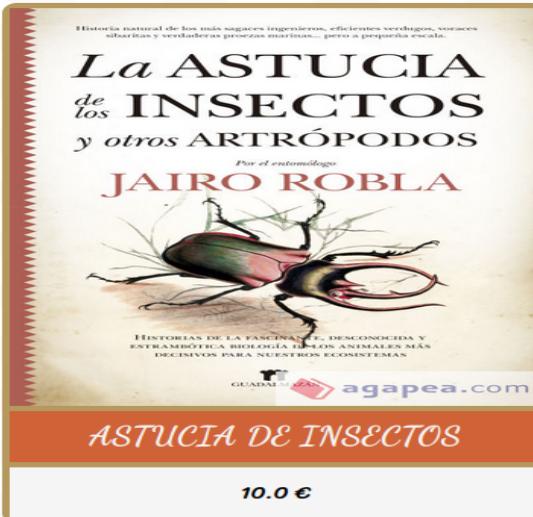
GÉNEROS

Aventura
 Biología
 Humor

Suspense
 Thriller

Aplicar
Resetear

Buscador...



Historia natural de los más sagaces ingenieros, eficientes verdugos, voraces sibaritas y verdaderas procesas maromas... pero a proporción escala.

La ASTUCIA de los INSECTOS y otros ARTRÓPODOS

Por el entomólogo **JAIRO ROBLA**

HISTORIAS DE LA FASCINANTE, DESCONOCIDA Y ENTRAMBIÓTICA BIOLOGÍA DE LOS ANIMALES MÁS DECISIVOS PARA NUESTROS ECOSISTEMAS

GUADALAJARA agapea.com

ASTUCIA DE INSECTOS

10.0 €

CATEGORÍAS

Libros
 Discos

GÉNEROS

Aventura
 Biología
 Humor

Suspense
 Thriller

Aplicar
Resetear

(*) Selecciona al menos un género

Iván Ayuso Olivera & Enrique Azorín Castellano

1º DAW - Proyecto intermodular - ARTES DORADAS

25

El filtro de la sección de productos funciona de la siguiente manera: de primeras, la sección de géneros se encontraría vacía, a no ser que se seleccione una de las categorías disponibles, en este caso libros o discos.

Una vez seleccionada una de las categorías, se cargarán los géneros correspondientes a cada una de estas. Una vez se ha realizado esta operación, el usuario podrá seleccionar el número deseado de géneros por el que quiera filtrar los diferentes productos.

Una vez seleccionados, si el usuario le da a “aplicar”, la página se recargará, mostrando solamente los productos que coincidan con los criterios de filtrado que han sido seleccionados.

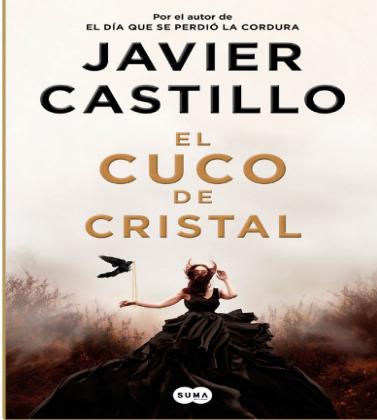
Un ejemplo sería que se seleccionarán de libros aquellos que sean de biología, haciendo que solamente se muestran los libros que tengan de género “biología”.

En el caso de que el usuario no seleccione ningún género y le de a enviar, se mostrará un mensaje de error que indicará al usuario: Selecciona al menos un género.



ARTES DORADAS

[Inicio](#) [Productos](#) [Sobre nosotros](#) [Contacto](#) [Iniciar sesión](#)



Por el autor de
EL DÍA QUE SE PERDIÓ LA CORDURA

JAVIER CASTILLO

EL CUCO DE CRISTAL

Nueva York, 2017. Cora Merlo, médico residente de primer año, sufre un infarto fulminante que la obliga a un trasplante de corazón. Aún convaleciente la joven recibe la visita de una extraña mujer con una enigmática oferta: pasar unos días en Steelville, un pequeño pueblo de interior, para conocer la vida de su hijo Charles, el donante de su corazón. Cora se adentra así en un hogar lleno de secretos, en un misterio que se extiende durante veinte años y en un pueblo hermético en el que, justo el día de su llegada, desaparece un bebé en un parque público.

AUTOR	Javier Castillo
EDITORIAL	Suma
FORMATO	Tapa blanda
Nº PÁGINAS	488
ISBN	9788491293052

Cuco de cristal

Autor: Javier Castillo

Editorial: Suma

[Añadir a deseados](#)

Stock: 13

Precio: 20.06 + IVA(21%)

[Añadir a la cesta](#)

OPIINIONES

[Deja una opinión de este producto](#)

(*) INICIA SESIÓN o REGÍSTRATE para dejar una reseña

(*) Realiza un pedido de este producto para dejar una reseña

Escribe una opinión...

[Enviar](#)

 **jlopez05** 20/01/2023 00:00:00

Me encantó el producto

[Términos y condiciones](#)
[Política de privacidad](#)

© 2022-23, Iván Ayuso & Enrique Azorín
1º DAW | IES Poeta Paco Mollá (Petrer)

Una vez que un usuario haya filtrado y encontrado el producto que buscaba en la sección de productos, se lo selecciona, será redirigido a una nueva sección, la vista de producto.

Esta nueva sección mostrará los datos de un producto en concreto, además de permitirle al usuario realizar ciertas acciones.

Desde esta vista, el usuario podrá:

- Añadir un producto a la cesta o carrito de la compra.
- Elegir la cantidad del producto que quiera añadir al carrito.
- Observar los diferentes datos que tiene el producto como puede ser: el stock, el autor o artista, la discográfica o editorial, una descripción del producto, diferentes datos alternativos como el número de páginas, canciones, asin, isbn, etc.
- Una sección de reseñas, donde se podrán ver las opiniones de otros usuarios que hayan comprado el producto.

OPINIONES

 **jlopez65** 20/01/2023 00:00:00

Me encantó el producto

(*) Aún no hay reseñas

La sección de opiniones de la vista de producto cargará las diferentes reseñas que se encuentran almacenadas en nuestra base de datos de un producto en específico.

Las opiniones mostrarán el usuario que la ha realizado, la fecha en la que se realizó y el mensaje dejado por el usuario, explicando que le ha parecido el producto.

En el caso de que no se hayan realizado opiniones todavía para un producto, el sistema mostrará un mensaje al usuario diciendo que: Aún no hay reseñas.

(*) INICIA SESIÓN o REGÍSTRATE para añadir a la lista de deseos o cesta

Hijos de la fabula

Otro de los errores que indica esta sección es que si un usuario anónimo intenta añadir un producto al carrito, se mostrará un mensaje de error que avisará al usuario de loguearse o registrarse: INICIA SESIÓN o REGÍSTRATE para añadir a la lista de deseados o cesta.

VER DATOS

LISTA DE DESEOS

CARRITO

PEDIDOS

FACTURAS

OPINIONES

LOG OUT

Otra de la página, vista o sección más significativa de la web es el perfil del usuario. Accesible solamente para clientes o administradores (un usuario anónimo no puede acceder a él), se trata de un índice en el que el usuario podrá visitar otras secciones que tengan que ver con sus datos personales.

Desde esta vista el usuario puede:

- VER DATOS: visitar su perfil en más detalle viendo todos sus datos.
- LISTA DE DESEOS: sin implementar.
- CARRITO: el usuario será redirigido a la sección del carrito.
- PEDIDOS: se actualizará la página para cargar una vista con todos los pedidos realizados por el usuario en formato tabla.
- FACTURAS: lo mismo que con los pedidos.
- OPINIONES: sin implementar.
- LOG OUT: se cerrará la sesión.

Esta vista será casi exactamente igual tanto para clientes como administradores, la única diferencia es que para los administradores no aparecerán los botones de “ver datos” que se sustituirá por “ver clientes”, y “carrito” y “lista de deseos”.



FOTO

NOMBRE
<i>Jlopez65</i>
CORREO ELECTRÓNICO
<i>Jlopez@gmail.com</i>
CONTRASEÑA
<i>11a03jd7fn</i>
TARJETAS
<i>40294758</i>
DIRECCIONES
<i>Avda. Comunidad Valenciana, 14</i> <i>Calle Andalucía, 04</i>
TELÉFONO
<i>6593020</i>
FECHA DE NACIMIENTO
<i>19/07/1965</i>
FECHA DE ÚLTIMA CONEXIÓN
<i>02/06/2023 a las 23:14:00</i>
TIPO
<i>Cliente</i>

La sección de ver datos del perfil mostrará todos los datos del usuario que se encuentre logueado en sesión.

Se mostrará su foto, su nombre, el correo electrónico, la contraseña, las tarjetas que tiene asociadas a la cuenta, todas las direcciones que ha añadido, su teléfono, la fecha de nacimiento, la fecha de última conexión y el tipo de usuario que es.

Esta sección será exclusiva para los clientes y no podrá ser accesible para los usuarios anónimos.

CLIENTE	NOMBRE	CORREO ELECTRÓNICO	CONTRASEÑA	FECHA ÚLTIMA CONEXIÓN	TELÉFONO	FECHA DE NACIMIENTO
3	jlopez65	jlopez@gmail.com	1la03jd7fn	02/06/2023 23:17	6593020	19/07/1965
4	manuelgarciacano	m.garcia.c@gmail.com	5ps01m2n4d	26/01/2023 00:00	6748620	03/12/1996
5	marieta95	mariacandel4312@gmail.com	p1owOaken5	28/01/2023 00:00	6532014	24/02/1995
6	97crislo	cristinalopez87@gmail.com	l0293je7fd	02/02/2023 00:00	6793882	06/02/1987
7	alex568	alex568@gmail.com	n384jd0a3l	31/01/2023 00:00	7220992	21/07/2000

En el caso de que sea un administrador el que acceda a esta sección, lo que se mostrará será una tabla con la información de todos los usuarios almacenados en la base de datos.



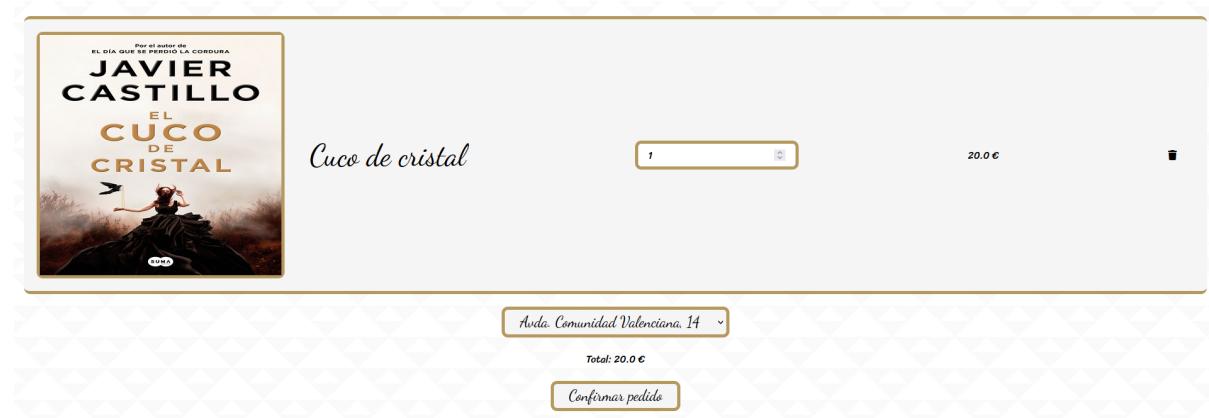
The screenshot shows a product page for the book "Cuco de cristal" by Javier Castillo. The page includes the book cover, the title, a quantity selector (set to 1), a price of 20.0 €, and a "Confirmar pedido" button. At the bottom, there are links for "Términos y condiciones" and "Política de privacidad".

La sección del carrito es otra de las vistas exclusivas para los usuarios clientes, ni los administradores puedan acceder a ella.

Esta sección lo que hará será mostrar al cliente los diferentes productos que haya ido añadiendo al carrito mientras navegaba por la sección de productos.



En el caso de que el cliente haya sido redirigido a esta sección y aún no haya añadido ningón carrito, se mostrará este mensaje de error: No tienes creado ningón carrito.



JAVIER CASTILLO
EL CUCO DE CRISTAL

Cuco de cristal

1 20.0 €

Avda. Comunidad Valenciana, 14

Total: 20.0 €

Confirmar pedido



JAVIER CASTILLO
EL CUCO DE CRISTAL

Cuco de cristal

2

40.0 €

Avda. Comunidad Valenciana, 14

Total: 40.0 €

[Confirmar pedido](#)

Desde esta sección, el cliente puede actualizar la cantidad de los diferentes productos que haya añadido al carrito, actualizando al cambio y modificando tanto el precio individual por producto como el total del pedido.

Así mismo, en cualquier momento el usuario puede borrar del carrito cualquier producto seleccionado el ícono de basura que se encuentra en la derecha.

Por último, el usuario puede seleccionar una dirección de pedido en el caso de que tenga varias, si no, siempre estará seleccionada la primera de las direcciones que ha insertado el usuario.

El cliente en cualquier momento puede confirmar el pedido, lo que realizará un insert en la base de datos y actualizará los pedidos de ese cliente, pero, en el caso de que no haya suficiente stock para poder realizar el pedido, el sistema mandará un mensaje de error: No tenemos suficiente stock para confirmar el pedido. Que impedirá al usuario realizar el pedido y evitará la inserción a la base de datos.

(*) No tenemos suficiente stock para confirmar el pedido



CÓDIGO DE PEDIDO	CÓDIGOS DE PRODUCTOS	DIRECCIÓN	FECHA	PRECIO TOTAL + IVA	FACTURADO	
1	1	Avda. Comunidad Valenciana, 14	15/01/2023 00:00	25,0 €	Sí	-
8	2	Avda. Comunidad Valenciana, 14	21/05/2023 18:28	17,0 €	Sí	-
9	3, 9	Avda. Comunidad Valenciana, 14	21/05/2023 18:29	34,0 €	Sí	-
10	7	Avda. Comunidad Valenciana, 14	21/05/2023 18:35	23,0 €	Sí	-
11	10	Avda. Comunidad Valenciana, 14	21/05/2023 19:49	21,0 €	Sí	-
12	8	Avda. Comunidad Valenciana, 14	21/05/2023 22:06	40,0 €	Sí	-
13	6	Avda. Comunidad Valenciana, 14	22/05/2023 15:15	35,0 €	Sí	-
14	6	Avda. Comunidad Valenciana, 14	22/05/2023 15:15	70,0 €	No	<div style="border: 1px solid #ccc; padding: 2px;">Avda. Comunidad Valenciana, 14</div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 2px;">Facturar</div>
15	1	Calle Andalucía, 04	02/06/2023 23:19	20,0 €	No	<div style="border: 1px solid #ccc; padding: 2px;">Calle Andalucía, 04</div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 2px;">Facturar</div>

CLIENTE	CÓDIGO DE PEDIDO	CÓDIGOS DE PRODUCTOS	DIRECCIÓN	FECHA	PRECIO TOTAL + IVA	FACTURADO	
jlopez@gmail.com	1	1	Avda. Comunidad Valenciana, 14	15/01/2023 00:00	25,0 €	Sí	-
m.garcia.c@gmail.com	2	6	Calle Mayor, 20	20/01/2023 00:00	20,0 €	Sí	-
mariacandel4312@gmail.com	3	10	Calle Blasco Ibáñez, 15 2C	21/01/2023 00:00	50,0 €	Sí	-
cristinalopez87@gmail.com	4	4	Plaza España, 9 3B	23/01/2023 00:00	65,0 €	Sí	-
alex568@gmail.com	5	8	Calle Santa Bárbara, 5	26/01/2023 00:00	35,0 €	Sí	-
mariacandel4312@gmail.com	6		Calle Blasco Ibáñez, 15 2C	27/01/2023 00:00	40,0 €	No	<div style="border: 1px solid #ccc; padding: 2px;">Calle Blasco Ibáñez, 15 2C</div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 2px;">Facturar</div>
alex568@gmail.com	7		Calle Santa Bárbara, 5	28/01/2023 00:00	15,0 €	Sí	-
jlopez@gmail.com	8	2	Avda. Comunidad Valenciana, 14	21/05/2023 18:28	17,0 €	Sí	-
jlopez@gmail.com	9	3, 9	Avda. Comunidad Valenciana, 14	21/05/2023 18:29	34,0 €	Sí	-
jlopez@gmail.com	10	7	Avda. Comunidad Valenciana, 14	21/05/2023 18:35	23,0 €	Sí	-
jlopez@gmail.com	11	10	Avda. Comunidad Valenciana, 14	21/05/2023 19:49	21,0 €	Sí	-
jlopez@gmail.com	12	8	Avda. Comunidad Valenciana, 14	21/05/2023 22:06	40,0 €	Sí	-
jlopez@gmail.com	13	6	Avda. Comunidad Valenciana, 14	22/05/2023 15:15	35,0 €	Sí	-
jlopez@gmail.com	14	6	Avda. Comunidad Valenciana, 14	22/05/2023 15:15	70,0 €	No	<div style="border: 1px solid #ccc; padding: 2px;">Avda. Comunidad Valenciana, 14</div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 2px;">Facturar</div>
jlopez@gmail.com	15	1	Calle Andalucía, 04	02/06/2023 23:19	20,0 €	No	<div style="border: 1px solid #ccc; padding: 2px;">Calle Andalucía, 04</div> <div style="border: 1px solid #ccc; padding: 2px; margin-top: 2px;">Facturar</div>

Una vez realizado el pedido, se redirige al usuario a la sección de pedidos, aunque tambien es posible acceder a ella mediante el enlace en el perfil.

Esta vista de la web, mostrará todos los pedidos realizados por el cliente, y en el caso de que sea un administrador, mostrará todos los pedidos realizados por todos los usuarios.

Desde esta sección, tanto clientes como administradores pueden actualizar los diferentes pedidos para cambiar su estado de “no facturado” a “facturado”.

El usuario puede elegir entre las diferentes direcciones que tenga, y puede elegir una opción diferente que la del propio pedido.

En el caso de que el usuario seleccione el botón facturar, el estado del pedido pasará a estar facturado y redirigirá al usuario a la sección de facturas.

CÓDIGO FACTURA	PEDIDO ASOCIADO	DIRECCIÓN FACTURACIÓN	FECHA DE FACTURACIÓN	TOTAL CON IVA	EXPORTAR A XML	EXPORTAR A PDF
1	1	Avda. Comunidad Valenciana, 14	15/01/2023 00:00	25.0 €		
6	8	Avda. Comunidad Valenciana, 14	21/05/2023 19:38	17.0 €		
7	10	Avda. Comunidad Valenciana, 14	21/05/2023 19:46	23.0 €		
8	9	Avda. Comunidad Valenciana, 14	21/05/2023 19:47	34.0 €		
9	11	Avda. Comunidad Valenciana, 14	21/05/2023 19:49	21.0 €		
10	12	Avda. Comunidad Valenciana, 14	22/05/2023 15:12	40.0 €		
12	13	Avda. Comunidad Valenciana, 14	28/05/2023 21:18	35.0 €		

La sección de facturas es muy parecida a la de pedidos, ya que tanto cliente como administrador tendrán datos diferentes.

Un cliente podrá ver todas sus facturas y un administrador podrá ver las facturas de todos los clientes.

La diferencia con pedidos, es que esta sección tendrá dos botones que servirán para exportar la factura a diferentes formatos, uno de ellos el XML y el otro en PDF.

CLIENTE	CÓDIGO FACTURA	PEDIDO ASOCIADO	DIRECCIÓN FACTURACIÓN	FECHA DE FACTURACIÓN	TOTAL CON IVA	EXPORTAR A XML	EXPORTAR A PDF
jlopez@gmail.com	1	1	Avda. Comunidad Valenciana, 14	15/01/2023 00:00	25.0 €		
m.garcia.c@gmail.com	2	2	Calle Mayor, 20	20/01/2023 00:00	20.0 €		
cristinalopez87@gmail.com	3	4	Plaza España, 9 3B	23/01/2023 00:00	65.0 €		
alex568@gmail.com	4	5	Calle Santa Bárbara, 5	26/01/2023 00:00	35.0 €		
alex568@gmail.com	5	7	Calle Santa Bárbara, 5	28/01/2023 00:00	15.0 €		
jlopez@gmail.com	6	8	Avda. Comunidad Valenciana, 14	21/05/2023 19:38	17.0 €		
jlopez@gmail.com	7	10	Avda. Comunidad Valenciana, 14	21/05/2023 19:46	23.0 €		
jlopez@gmail.com	8	9	Avda. Comunidad Valenciana, 14	21/05/2023 19:47	34.0 €		
jlopez@gmail.com	9	11	Avda. Comunidad Valenciana, 14	21/05/2023 19:49	21.0 €		
jlopez@gmail.com	10	12	Avda. Comunidad Valenciana, 14	22/05/2023 15:12	40.0 €		
mariacondel4312@gmail.com	11	3	Calle Blasco Ibáñez, 15 2C	26/05/2023 15:57	50.0 €		
jlopez@gmail.com	12	13	Avda. Comunidad Valenciana, 14	28/05/2023 21:18	35.0 €		

```

Factura1(1).xml  X

<?xml version="1.0" encoding="utf-8"?>
<factura codigo_factura="1">
    <fecha>
        <dia>15</dia>
        <mes>1</mes>
        <año>2023</año>
        <hora>0</hora>
        <minuto>0</minuto>
        <segundo>0</segundo>
    </fecha>
    <direccion_facturacion>
        <direccion>Avda. Comunidad Valenciana, 14</direccion>
        <cp>3640</cp>
        <poblacion>Monóvar</poblacion>
        <provincia>Alicante</provincia>
    </direccion_facturacion>
    <pedido_asociado codigo_pedido="1">
        <cliente>jlopez65</cliente>
        <direccion_envio>
            <direccion>Avda. Comunidad Valenciana, 14</direccion>
            <cp>3640</cp>
            <poblacion>Monóvar</poblacion>
            <provincia>Alicante</provincia>
        </direccion_envio>
        <lineas>
            <producto codigo_producto="1">
                <nombbre>Cuco de cristal</nombbre>
                <precio_unitario>20.0</precio_unitario>
                <iva>21</iva>
                <cantidad_productos>1</cantidad_productos>
                <total_linea>10.0</total_linea>
            </producto>
        </lineas>
    </pedido_asociado>
    <base_imponible>25.0</base_imponible>
    <total>25.0</total>
</factura>

```

Desde la sección de productos la factura puede ser exportada a XML y este es un resultado de ejemplo, en el que cada uno de los campos de la factura es una etiqueta en el XML.

Factura nº 1

Datos del cliente:

Nº cliente: 3
Nombre: jlopez65
E-mail: jlopez@gmail.com
Teléfono: 6593020

Datos de la factura:

Fecha: 15-01-2023 00:00:00
Dirección de facturación: Avda. Comunidad Valenciana, 14; 3640; Monóvar, Alicante
Dirección de envío: Avda. Comunidad Valenciana, 14; 3640; Monóvar, Alicante

Líneas del pedido:

CÓDIGO	NOMBRE	PRECIO	I.V.A.	CANTIDAD	IMPORTE
1	Cuco de cristal	20,00 €	21%	1	20,00 €

BASE IMPONIBLE: 25,00 €
TOTAL FACTURA: 25,00 €

También está la posibilidad de exportar la factura a PDF.

La factura se descargará en un formato que mostrará los datos más significativos del cliente y de la propia factura, además de añadir todos los productos que el cliente ha comprado, es decir la línea del pedido, mostrando también el importe total de la factura.

(*) No tienes permisos para acceder a esta sección

Por último, todas y cada una de las secciones que solamente son accesibles para un cliente o un administrador, en caso de que un usuario anónimo intente acceder a ellas, se mostrará un mensaje de error que diga: No tienes permisos para acceder a esta sección.

4) LENGUAJE DE MARCAS

a) Mockup

Figma:

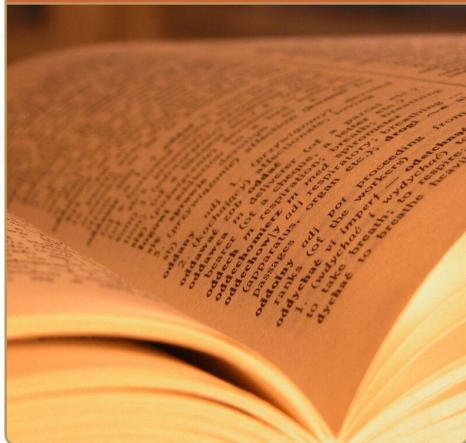
https://www.figma.com/file/QxGluLgWCC2bVUDivRd4T8/Tienda_Online?type=design&node_id=0%3A1&t=hTJk6et2NRwb5ykF-1

El diseño de nuestra aplicación no ha cambiado mucho, al final nos hemos quedado con el siguiente diseño final:

El header y el footer serán el mismo para todas la páginas. El header tendrá dos secciones una el título y otra el navegador.



LIBROS



DISCOS



El índice o página principal dispondrá de dos tarjetas, divididas por la mitad de la página que mostrará al cliente nuestras dos ofertas de productos, libros o discos, y que servirán de enlace a la página de producto.

LOGIN

CORREO:

CONTRASEÑA:

El formulario de login tendrá un diseño simple, donde el usuario insertará su correo electrónico y su contraseña.

VER DATOS

LISTA DE DESEOS

CARRITO

PEDIDOS

FACTURAS

OPINIONES

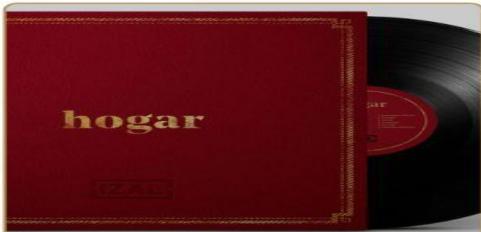
LOG OUT

La página de perfil será la siguiente, donde el usuario podrá visualizar diferentes botones para ir a sus respectivas páginas con sus datos personales como ver datos, lista de deseos, etc.



La página de productos dispondrá de dos secciones, una el filtro, y otra que contendrá todos los productos de nuestra tienda.





Quisque ipsum dolor sit amet, consectetur adipiscing elit. Quisque ultricies volutpat ultricies. Nunc malesuada dui felis, quis mollis sapien laoreet eu. Donec vitae eleifend dui. Vestibulum nec malesuada odio. In hac habitasse platea dictumst. Morbi efficitur, magna at viverra convallis, erat orci congue mauris, et laoreet sem diam vel libero. Maecenas nec dolor nibh. Morbi at nibh vitae nibh lobortis dictum sed in dui. Quisque rutrum suscipit rhoncus. Pellentesque convallis risus quis turpis tincidunt, faucibus luctus mauris lobortis. Cras sit amet leo efficitur, feugiat nisi eget, egestas tortor.

TÍTULO DATO 1	DATO 1
TÍTULO DATO 2	DATO 2
TÍTULO DATO 3	DATO 3
TÍTULO DATO 4	DATO 4
TÍTULO DATO 5	DATO 5

OPINIONES

Deja una opinión de este producto

Formulario mensaje...

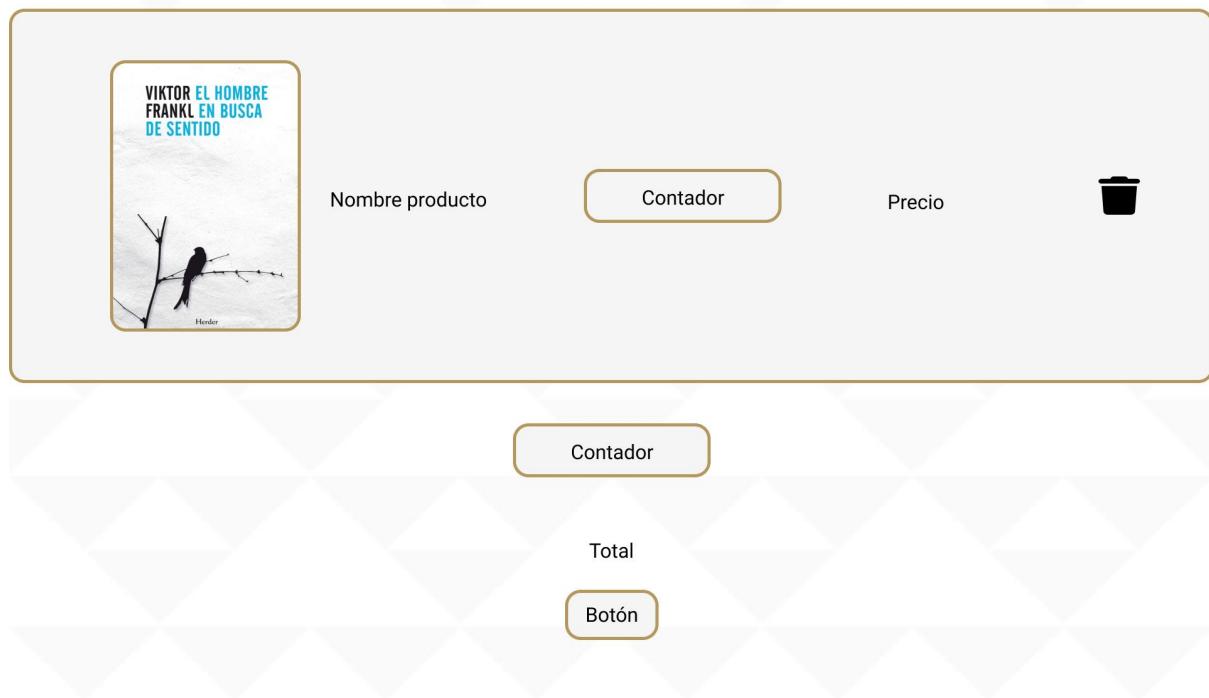
OPINIÓN

Quisque ipsum dolor sit amet, consectetur adipiscing elit. Quisque ultricies volutpat ultricies. Nunc malesuada dui felis, quis mollis sapien laoreet eu. Donec vitae eleifend dui. Vestibulum nec malesuada odio. In hac habitasse platea dictumst. Morbi efficitur, magna at viverra convallis, erat orci congue mauris, et laoreet sem diam vel libero. Maecenas nec dolor nibh. Morbi at nibh vitae nibh lobortis dictum sed in dui. Quisque rutrum suscipit rhoncus. Pellentesque convallis risus quis turpis tincidunt, faucibus luctus mauris lobortis. Cras sit amet leo efficitur, feugiat nisi eget, egestas tortor.

La página del producto será la más extensa de todas, donde el cliente observará todos los datos del producto, dividido de la siguiente manera: datos comunes para todos los productos, datos únicos del producto y las opiniones que tiene este producto.



La cesta o carrito tendrá un diseño que muestre en secciones los diferentes productos añadidos a esta, en el que mostrará su foto, el nombre del producto, la cantidad, el precio y un botón para poder eliminarlo del carrito.



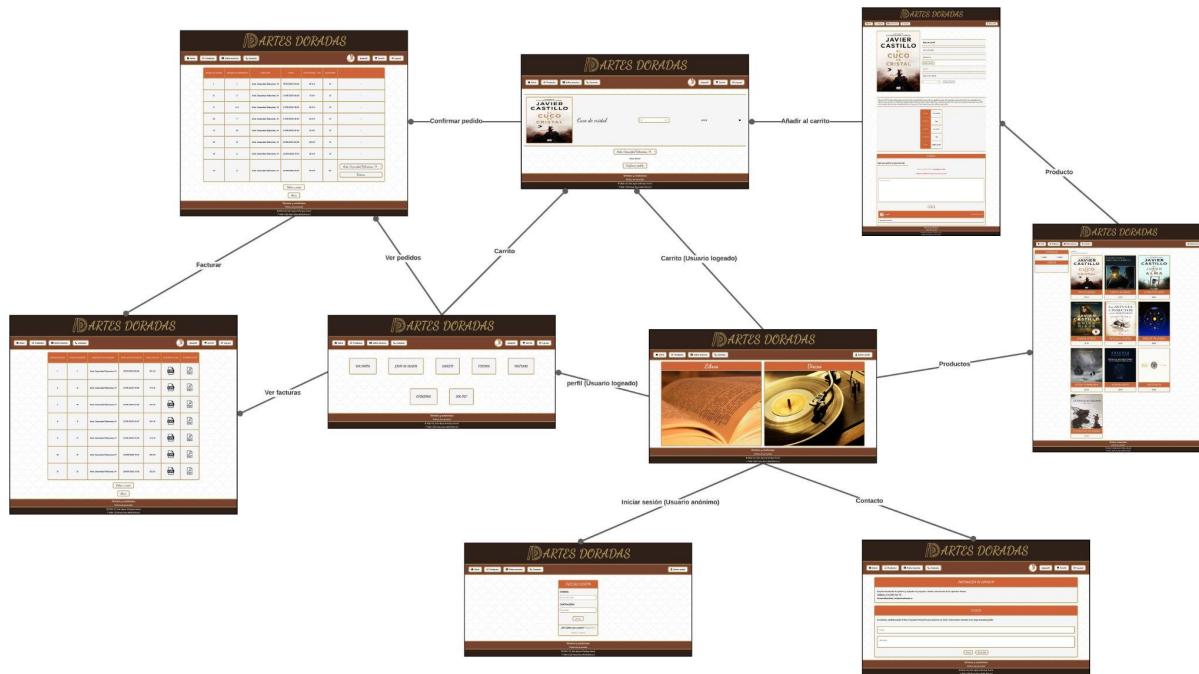
| TÍTULO DE DATO |
|----------------|----------------|----------------|----------------|----------------|
| DATO | DATO | DATO | DATO | DATO |
| DATO | DATO | DATO | DATO | DATO |
| DATO | DATO | DATO | DATO | DATO |
| DATO | DATO | DATO | DATO | DATO |
| DATO | DATO | DATO | DATO | DATO |

Por último, esta será la sección que mostrará todos los datos de un cliente, como pueden ser sus pedidos, facturas, etc.

b) Diagrama de flujo de pantallas

Lucidchart:

https://lucid.app/lucidchart/6f31b2c0-002f-414c-a9ca-8a21bf931ab9/edit?viewport_loc=-4937%2C-2182%2C9469%2C4729%2C0_0&invitationId=inv_be500f78-5005-4244-9bdc-5553c2eecbff



5) CONCLUSIONES Y DIFICULTADES ENCONTRADAS

Uno de los principales problemas ha sido la base de datos, en concreto la estructura del modelo entidad / relación. El principal problema que hemos encontrado con el diagrama, ha sido la actualización constante de su estructura, eliminaciones, recortes o actualizaciones.

Conforme íbamos avanzando con el proyecto, en muchas ocasiones nos dimos cuenta de que la estructura del esquema era errónea o se podría mejorar para facilitar algunas funcionalidades en java. También, el diagrama en muchas ocasiones nos ocultaba el traspaso de datos desde SQL a objetos de java. En varias ocasiones tuvimos que modificar tanto el esquema entidad / relación como los propios DDL y DML para así poder realizar un mayor trabajo desde el lenguaje Java, aunque en muchas ocasiones decidimos complicar más la labor del controlador en java que retocar la base de datos ya que las modificaciones que pudiéramos crear pudieran complicar aún más las cosas.

Otra dificultad con la que nos encontramos fue la falta de tiempo. Al principio, cuando estábamos comenzando con el proyecto e íbamos poco a poco creando su estructura, esto no supuso un problema. Pero, conforme íbamos avanzando y sobre todo al empezar el trabajo en Java, el tiempo disponible para el proyecto empezó a ser un problema, y por la falta de este, tuvimos que implementar las funcionalidades justas que se pedía para el proyecto.

Esto supuso que muchas de las funcionalidades que teníamos plasmadas ya en el modelo entidad / relación no salieran a la luz en el proyecto final, y que muchas de ellas se quedaran a la espera de que se implementara dicha funcionalidad, como pueden ser las listas de deseos, las opiniones, etc.

También respecto a Java, tuvimos el problema de que se nos complicó mucho la estructura del modelo controlador, el dto y el dao, el response y el request, etc. Ya que era la primera vez que lo veíamos y al principio perdimos mucho tiempo intentando entender cómo funcionaban los servlets e intentar entender la estructura de estos.

Dejando de lado la programación, y centrándonos más en el lenguaje de marcas, el diseño de la web también tuvo su serie de problemas. Conforme íbamos avanzando en el proyecto, el diseño cambió a lo largo del desarrollo de este, bien por el rediseño de algunas de las páginas para mejorar la visualización o por el cambio completo de estructura de flex a grid.

Debido también por la falta de tiempo por el tramo final del proyecto, muchas páginas son informaciones en tablas en vez de un posible diseño que teníamos anteriormente más visual y no tan simple como puede ser una tabla de datos.

Dejando de lado los problemas, el proyecto nos ha parecido una experiencia muy enriquecedora y que nos ha ayudado a mejorar en muchos aspectos:

-
- Mejoras en el uso del HTML y el CSS: la distribución de la página, su estructura interna y su visualización, han cambiado mucho a lo largo del desarrollo del proyecto, y conforme íbamos avanzando hemos ido añadido mejoras visuales y de estructura para poder hacer la página mucho más agradable y cómoda, como el diseño adaptable a varias pantallas y una mejor visualización de los contenidos para los diferentes tipos de usuarios.
 - Entendimiento del conjunto global del uso de HTML, CSS, SQL y Java para la realización de una aplicación web: al principio del curso esta era una de las grandes dudas que teníamos y era como se iba a conectar todo para poder realizar una tienda online. Es decir, al principio íbamos poco a poco diseñando la estructura de la base de datos, el diseño de la web, etc...y no fue hasta llegar a programación y empezar a trabajar con los servlets y los jsp que poco a poco empezamos a ver el conjunto global de la aplicación. Ver todo en conjunto gracias a programación nos ha ayudado a pensar en las otras partes desde una nueva perspectiva, ya que conforme íbamos avanzando nos íbamos dando cuenta de las implicaciones que tenía haber realizado las relaciones en el entidad / relación de una forma o el diseño de la página web de otra. Y gracias a ello, ahora, a la hora de realizar de nuevo un esquema entidad / relación o diseñar de nuevo la web, lo veríamos desde otra perspectiva y cambiaríamos cosas o añadiríamos atributos o relaciones que antes no se nos hubieran ocurrido.
 - Experiencia: una de las grandes ventajas que tiene haber realizado este proyecto ha sido poder saber como sería realizar o al menos, lo más próximo a realizar un proyecto fuera de los trabajos del ciclo. La realización de una tienda online nos ha servido para poder entender o tener al menos unos conocimientos básicos de cómo sería realizar esta clase de proyectos de una manera más global y en un equipo, en vez de realizar solamente proyectos individuales y más pequeños.

6) POSIBLES MEJORAS O AMPLIACIONES

Debido a la falta de tiempo, hay muchas funcionalidades que sí que plasmamos en el entidad / relación pero que no pudimos realizar al final del proyecto.

Una de estas funcionalidades sería la barra de búsqueda de la sección de productos. En ella, el usuario podría escribir el nombre de cualquiera de nuestros productos para poder encontrarlo de una manera más sencilla que con el filtro.



La funcionalidad del buscador iría a un servlet específico que recogería la cadena de texto introducido por el usuario en la barra de búsqueda y realizará una búsqueda en la base de datos para comprobar si hay algún producto coincidente, y de ser así el caso mostrárselos al usuario en la vista de productos.

Otra mejora sería la implementación de la lista de deseos. En el entidad / relación hay plasmadas tablas para realizar esta funcionalidad pero no implementada en java mediante los servlets y los jsp.

El añadido sería realizar un servlet para que el usuario pudiera añadir un producto a la lista de deseos implementada en la base de datos mediante inserts.

Además, necesitamos realizar una nueva vista para poder observar de manera visual la lista de deseos de un usuario.



Editorial: Suma

Añadir a deseados

Stock: 13



User profile icons: *jlopez65*, *Carrito*, *Log out*

ATOS

LISTA DE DESEOS

CA

Otra funcionalidad que también está plasmada actualmente en la base de datos pero no se realiza mediante programación, es la capacidad de un usuario para poder escribir opiniones sobre los diferentes productos que haya comprado.

Actualmente en la página de producto se puede observar un formulario y las diferentes opiniones de algunos usuarios. El formulario actualmente no está operativo, pero las opiniones si que son aquellas que tenemos registradas en la base de datos.



OPINIONES

Deja una opinión de este producto

() Deja una opinión de este producto para dejar una revisión*

() Deja una opinión de este producto para dejar una revisión*

Escribe una opinión...

Cancelar

Por último, algunas de las ampliaciones según el tipo de usuario serían las siguientes:

- Administradores:
 - Capacidad para poder añadir y modificar productos desde la página web.
 - Añadir la funcionalidad para que se pueda editar la información de un cliente.
- Clientes:
 - Implementar un formulario para que un usuario anónimo pueda registrarse como un cliente, para poder acceder a las funcionalidades de la web.

7) BIBLIOGRAFÍA

- **AUTORES**

- Iván Ayuso Olivera: <https://github.com/IvanAO99>
- Enrique Azorín Castellano: <https://github.com/Kikeaks>

- **RECURSOS UTILIZADOS**

- Figma: <https://www.figma.com/>
- Font Awesome: <https://fontawesome.com/>
- GitHub: <https://github.com/>
- Lucidchart: <https://lucid.co/>

- **TECNOLOGÍAS UTILIZADAS**

- Apache NetBeans: <https://netbeans.apache.org/>
- Docker: <https://www.docker.com/>
- Git: <https://git-scm.com/>
- Java JDK:
<https://www.oracle.com/es/java/technologies/downloads/#jdk17-windows>
- Oracle SQL Developer: <https://www.oracle.com/es/database/sqldeveloper/>
- Visual Studio Code: <https://code.visualstudio.com/>

- **TEORÍA**

- Bases de datos
 - Unidad 2. Diseño de bases de datos
 - Unidad 3. Definición de datos en SQL
 - Unidad 4. Manipulación de datos en SQL
 - Unidad 5. Consultas en SQL
- Lenguaje de marcas
 - Unidad 2. HTML5; Características principales
 - Unidad 3. CSS3; Características principales
 - Unidad 4. HTML5, CSS3 & JavaScript; Características avanzadas
 - Unidad 5. Documentos XML

-
- Programación
 - Unidad 11. Conexión con bases de datos relacionales
 - Unidad 12. Introducción a la programación web con Java
 - **TUTORES**
 - José Ramón Mas Davó
 - Silvia Amorós Hernández
 - Miguel Ángel Tomás Amat