



PY01

Gestión de Librería

David Calvo, Erin Camacho, Johnsy López

Instituto Tecnológico de Costa Rica

IC4700 Lenguajes de Programación

Prof. Allan Rodríguez Dávila

II Semestre

16 de septiembre del 2025

Manual de Usuario

1. Instalar Compilador Embarcadero

- Ingresa a <https://www.embarcadero.com/es/free-tools/ccompiler/free-download/thank-you> y descargar el .zip.
- En la ubicación del archivo, clic derecho sobre el archivo .zip descargado y selecciona “Extraer aquí”.
- Presiona Win + R, escribe sysdm.cpl, ve a Opciones Avanzadas y Variables de entorno.
- En la sección Variables del Sistema, edita la variable Path.
- Agrega la ruta a la carpeta bin del compilador previamente extraído.
- Acepta todos los cambios.

2. Clonar repositorio github

- Desde el Símbolo del sistema (cmd), usar el comando “git clone <https://github.com/johnsydev/p1-lenguajes-gestion-de-libreria#>”
- Escribe cd programa
- Escribe este comando: bcc32c -o programa main.c menu.c datos.c auxiliares.c config.c y luego .\programa.exe
- Se abrirá el menú principal del sistema en la consola.

3. Menu Principal y Uso General

```
PS C:\Users\garci\OneDrive\Desktop\p1-lenguajes-gestion-de-libreria\programa> bcc32c -o programa main.c menu.c datos.c auxiliares.c config.c
Embarcadero C++ 7.30 for Win32 Copyright (c) 2012-2017 Embarcadero Technologies, Inc.
main.c:
menu.c:
datos.c:
auxiliares.c:
config.c:
Turbo Incremental Link 6.90 Copyright (c) 1997-2017 Embarcadero Technologies, Inc.
PS C:\Users\garci\OneDrive\Desktop\p1-lenguajes-gestion-de-libreria\programa> .\programa.exe
----- Menu Principal -----

1. Administrativo
2. Opciones principales
3. Salir
```

- Al iniciar, el sistema cargará automáticamente los datos desde los archivos de texto (si existen).
- El menú principal muestra las opciones **Administrativas** y **Generales**.
- Para acceder a las opciones administrativas se debe ingresar el **usuario y contraseña** almacenados en el archivo correspondiente (config.json).
- El sistema guardará los cambios realizados (libros, clientes, pedidos) en los archivos de texto al salir, asegurando la persistencia de la información.
- Para mayor explicación de funcionalidades, observa el siguiente apartado.

Pruebas de Funcionalidad

```
----- Menu Principal -----
```

1. Administrativo
2. Opciones principales
3. Salir

Al iniciar el programa se muestra este menú, con opción Administrativo, Opciones Generales y Salir.

A) Administrativo

```
Seleccione una opcion: 1
```

```
----- Autenticacion -----
```

```
Nombre de usuario: admin
```

```
Contraseña: admin123
```

```
Acceso concedido.
```

Para ingresar a esta opción, se solicitará usuario y contraseña de la librería, en este caso definidos como admin y admin123 respectivamente.

```
----- Menu Administrativo -----
```

1. Registrar libros
2. Eliminar Libro
3. Manejo de inventario
4. Registrar clientes
5. Eliminar Cliente
6. Crear pedido
7. Modificar Pedido
8. Eliminar Pedido
9. Consulta de pedidos
10. Estadísticas
11. Salir

```
Seleccione una opcion: █
```

Al tener acceso concedido se mostrará este menú con esta variedad de opciones.

1) Registrar Libro

```
----- Agregar Libro -----
```

```
Codigo del libro:DL1
```

```
Nombre del libro: Buscar Tesoros
```

```
Autor: David Perez
```

```
Precio: 10000
```

```
Cantidad disponible: 20
```

```
Libro agregado correctamente.
```

Al agregar un libro se solicitarán los siguientes datos: Código de libro único, título y autor como caracteres, precio flotante y cantidad entero, ambos positivos,

```
----- Agregar Libro -----
```

```
Codigo del libro:DL1
```

```
Nombre del libro: D
```

```
Autor: AX
```

```
Precio: 12
```

```
Cantidad disponible: 5
```

```
Error: Ya existe un libro con ese codigo.
```

Si se intenta colocar otro libro con el mismo código, lanza este error.

2) Eliminar Libro

```
----- Eliminar Libro -----
```

```
Codigo del libro: DL1
```

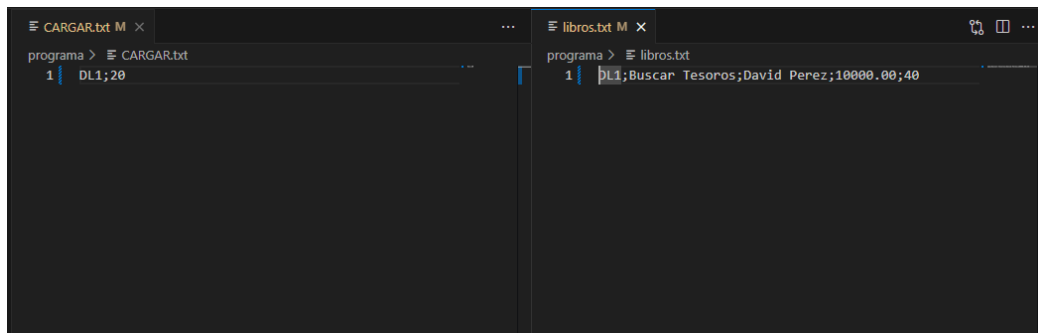
```
No se puede eliminar: el libro aparece en pedidos.
```

Permite eliminar un libro siempre y cuando este no se encuentre asociado a un pedido.

3) Manejo de Inventario

```
----- Manejo de inventario -----  
  
Este menu permite modificar la cantidad de libros disponibles, cargando la informacion desde un archivo.  
  
Escriba el nombre de archivo o presione [Enter] para volver atras.  
  
> CARGAR.txt
```

Este apartado permite dar mantenimiento al stock de libros por medio de una carga de lote, utilizando el archivo CARGAR.txt. Suma o resta el stock existente,



4) Registrar clientes

```
----- Registrar Cliente -----  
  
Cedula: 703260035  
Nombre: David Calvo  
Telefono: 87334321  
Cliente registrado correctamente.
```

Al registrar un cliente se solicitará su cedula única, y nombre y teléfono como cadena de caracteres.

```
----- Registrar Cliente -----  
  
Cedula: 703260035  
Nombre: De  
Telefono: 88888888  
Error: Ya existe un cliente con esa cedula.  
  
Cliente registrado correctamente.
```

Si se intenta registrar otro cliente con la misma cedula lanza este error.

5) Eliminar Cliente

```
----- Eliminar Cliente -----  
  
Cedula del cliente: 703260035  
No se puede eliminar: el cliente tiene pedidos asociados.
```

Permite eliminar un cliente registrado, siempre y cuando no esté asociado a un pedido.

6) Crear pedido

```
----- Crear Pedido -----  
  
=== GESTION DE PEDIDO ===  
1. Agregar linea  
2. Eliminar linea  
3. Mostrar pedido actual  
4. Generar pedido  
5. Salir sin guardar  
  
Seleccione una opcion: █
```

Para la creación del pedido se muestra el siguiente menú con dichas opciones como si fuera un bucle. Es el mismo pedido hasta que se genere o se salga sin guardar.

6.1 Agregar Línea

```
Seleccione una opcion: 1  
  
Filtrar por autor (ENTER para todos):  
  
--- Catalogo disponible ---  
No.  Codigo      Nombre      Precio  
-----  
1    DL1         Buscar Tesoros  $10000.00  
  
Ingrese el codigo del libro a agregar (o ENTER para cancelar): DL1  
Cantidad: 20  
Linea agregada correctamente.
```

Básicamente se muestra el catálogo de libros, se escoge el libro y la cantidad comprada para el respectivo pedido,

6.2 Eliminar línea

```
=== DETALLE DEL PEDIDO ===  
No. Codigo      Nombre      Precio  Cant.  Subtotal  
-----  
1   DL1         Buscar Tesoros  $10000.00  20    $200000.00  
  
Numero de linea a eliminar: 1  
Linea eliminada correctamente.
```

Se escoge el número de línea y se elimina dicha línea agregada del pedido.

6.3 Mostrar pedido actual

```
=== DETALLE DEL PEDIDO ===
No. Codigo      Nombre                      Precio  Cant.   Subtotal
-----
1  DL1          Buscar Tesoros              $10000.00 20    $200000.00

Subtotal: $200000.00
Impuesto (13%): $26000.00
Total: $226000.00
```

Se muestra el detalle del pedido, en el caso de no haber líneas se muestra el mensaje “No hay líneas en el pedido”.

6.4 Generar pedido

```
Cedula del cliente: 703260035
Fecha del pedido (dd/mm/yyyy): 15/9/2025

=====
Libreria La Alegria
Tel: 2234-5678
Cúdula: 3-101-123456
=====

Pedido No: 1000
Fecha: 15/9/2025
Cliente: David Calvo (703260035)
=====

=== DETALLE DEL PEDIDO ===
No. Codigo      Nombre                      Precio  Cant.   Subtotal
-----
1  DL1          Buscar Tesoros              $10000.00 20    $200000.00

-----
Subtotal:      $200000.00
Impuesto (13%): $26000.00
TOTAL:         $226000.00
=====

Horario: Lunes a Viernes: 8:00 AM - 6:00 PM, Sábados: 8:00 AM - 4:00 PM
=====

Pedido generado exitosamente.
```

Se solicita la cedula (existente) del cliente, y la fecha del pedido. Posteriormente, se muestra el detalle del pedido completo.

Si el pedido no tiene líneas, no es posible generar el pedido,

Se actualiza el stock del libro.

6.5 Salir sin guardar

Básicamente, no se genera ni guarda ninguna línea del pedido.

7) Modificar Pedido

```
----- Modificar Pedido -----  
  
ID          Cliente          Fecha          Total  
-----  
1000        David Calvo        15/9/2025      $226000.00  
  
Identificador del pedido a modificar: 1000  
  
=== EDITAR PEDIDO 1000 ===  
=== DETALLE DEL PEDIDO ===  
No.Codigo    Nombre          Precio    Cant.    Subtotal  
-----  
1  DL1        Buscar Tesoros    $10000.00  20      $200000.00  
  
1. Cambiar cantidad de una línea  
2. Agregar línea  
3. Eliminar línea  
4. Guardar cambios  
5. Cancelar sin guardar  
  
Seleccione una opcion: █
```

Para la modificación del pedido se muestra los pedidos existentes, se solicita que se escoja uno y luego un menú con dichas opciones como si fuera un bucle. Es el mismo pedido hasta que se guarden los cambios o se cancele sin guardar.

7.1 Cambiar cantidad de una línea

```
Seleccione una opcion: 1  
Numero de línea: 1  
Cantidad actual: 20  
Nueva cantidad: 30  
Línea actualizada.  
  
=== EDITAR PEDIDO 1000 ===  
=== DETALLE DEL PEDIDO ===  
No.Codigo    Nombre          Precio    Cant.    Subtotal  
-----  
1  DL1        Buscar Tesoros    $10000.00  30      $300000.00
```

Se muestra la cantidad actual y se solicita una nueva cantidad. Esta nueva cantidad se suma o resta la diferencia con la actual, en el stock del libro. Si el pedido tenía 20 y se actualiza a 30, el stock del libro se resta 10 y etc.

7.2 Agregar línea

```
Seleccione una opcion: 2  
Codigo del libro: DL1  
Cantidad: 2  
Línea agregada correctamente.  
  
=== EDITAR PEDIDO 1003 ===  
=== DETALLE DEL PEDIDO ===  
No.Codigo    Nombre          Precio    Cant.    Subtotal  
-----  
1  E4        Bing            $5000.00  23      $115000.00  
2  DL1        Busca Tesoros    $10000.00  2       $20000.00
```

Se agrega una nueva línea al pedido, lo mismo que con crear pedido.

7.3 Eliminar Línea

```
Seleccione una opcion: 3
Numero de linea a eliminar: 2
Linea eliminada correctamente.

=== EDITAR PEDIDO 1003 ===
=== DETALLE DEL PEDIDO ===
No.  Codigo      Nombre      Precio  Cant.  Subtotal
-----
1    E4          Bing       $5000.00 23     $115000.00
```

Eliminar línea de un pedido básicamente elimina esa línea del pedido.

7.4 Guardar Cambios

Pedido actualizado correctamente.

7.5 Cancelar sin guardar

No se efectúa ningún cambio.

8) Eliminar Pedido

```
----- Eliminar Pedido -----

Pedidos disponibles:
ID      Cliente      Fecha      Total
-----
1003    David Calvo      16/09/2025  $129950.00
1004    David Calvo      10/09/2024  $113000.00
1005    David Calvo      16/09/2025  $11300.00

Identificador del pedido a eliminar: 1003
Stock restaurado para libro E4: +23 unidades
Pedido eliminado correctamente y stock restaurado.
```

Se elimina un pedido creado, y se restaura el respectivo stock del libro (aumentar cantidad).

9) Consulta de pedidos

```
----- Consulta de pedidos -----

No.  Identificador  Cliente      Fecha      Subtotal  Total
-----
1    1004          David Calvo  10/09/2024  $100000.00 $113000.00
2    1005          David Calvo  16/09/2025  $10000.00  $11300.00
```

Se realiza la consulta de pedidos existentes.

```

Seleccione un pedido por numero (1-2) o 0 para volver: 1

=====
                      Libreria La Alegria
                      Tel: 2234-5678
                      C dula: 3-101-123456
=====
Pedido No: 1004
Fecha: 10/09/2024
Cliente: David Calvo (703260035)
=====

=== DETALLE DEL PEDIDO ===
No. Codigo      Nombre                Precio  Cant.    Subtotal
-----
1   DL1         Busca Tesoros                $10000.00  10    $100000.00
-----

Subtotal:      $100000.00
Impuesto (13%): $13000.00
TOTAL:         $113000.00
=====
Horario: Lunes a Viernes: 8:00 AM - 6:00 PM, S bados: 8:00 AM - 4:00 PM
=====

Presione ENTER para volver a la lista...

```

Si se desea, se puede seleccionar un pedido por n mero de l nea para conocer el detalle completo del pedido.

10) Estadisticas

```

----- Estadisticas -----

1. Total de ventas por anio
2. Total de ventas por mes/anio
3. Clientes con mas pedidos
4. Libros mas vendidos
5. Autor con mas ventas por anio
6. Volver

Seleccione una opcion:

```

El men  de estad sticas tiene esta variedad de opciones.

10.1 Total de Ventas por A o

```

Seleccione una opcion: 1
=== TOTAL DE VENTAS ===
Pedidos totales: 2
Monto total:      $124300.00

=== MONTO POR ADO ===
A o      Pedidos    Monto
-----
2024      1        $113000.00
2025      1        $11300.00

Presione ENTER para continuar...

```

Se muestra el total generado por cada pedido por a o.

10.2 Total de ventas por mes/año

```
Seleccione una opcion: 2
=== TOTAL DE VENTAS POR MES-ANIO ===
Mes/Año  Pedidos  Monto
-----
09/2024   1      $113000.00
09/2025   1      $11300.00

Presione ENTER para continuar...
```

Se muestra el total agrupado por mes/año.

10.3 Clientes con más pedidos

```
Seleccione una opcion: 3
Cuantos desea ver (Top N): 10
=== CLIENTES CON MAS PEDIDOS (TOP 10) ===
No.  Cedula      Nombre      Pedidos  Monto
-----
1    703260035    David Calvo    2      $124300.00

Presione ENTER para continuar...
```

Se muestran los clientes con más pedidos, mostrando su cedula, nombre, pedidos y monto.

10.4 Libros más vendidos

```
Seleccione una opcion: 4
Cuantos desea ver (Top N): 10
Filtrar por año (ej: 2025) o ENTER para todos:
=== LIBROS MAS VENDIDOS (TOP 10) ===
No. Codigo      Titulo      Cantidad
-----
1    DL1          Busca Tesoros    10
2    E4          Bing             2

Presione ENTER para continuar...
```

Se muestran los libros más vendidos, mostrando el código, título y cantidad vendida.

```
Filtrar por año (ej: 2025) o ENTER para todos: 2024
=== LIBROS MAS VENDIDOS (TOP 10) - Año 2024 ===
No. Codigo      Titulo      Cantidad
-----
1    DL1          Busca Tesoros    10

Presione ENTER para continuar...
```

Existe el filtro opcional que permite filtrar esos libros por año del pedido.

10.5 Autor con más ventas por año

```
Seleccione una opcion: 5
=== AUTOR CON MAS VENTAS POR ANIO (POR MONTO) ===
Anio      Autor          Unidades  Monto
-----
2024      Pepe                10      $100000.00
2025      RK                   2      $10000.00
```

Se muestran los autores con mas ventas por año, mostrando el año, autor, unidades y monto.

11) Salir

Regresa al menú principal.

B) Opciones Generales

```
----- Opciones Principales -----
1. Consulta de catalogo
2. Consulta de cliente
3. Salir
Seleccione una opcion: |
```

Este apartado es de acceso público, para cualquier usuario, no requiere de ningún acceso con usuario/contraseña. Contiene la consulta de catálogo, consulta de cliente y Salir.

1) Consulta de Catalogo

```
----- Catalogo -----
-----
Codigo: E4
Nombre: Bing
Autor: RK
Precio: 5000.00
Cantidad: 23
-----
Codigo: DL1
Nombre: Busca Tesoros
Autor: Pepe
Precio: 10000.00
Cantidad: 10
-----
```

Se muestra el catálogo actual de la librería, los libros en stock.

2) Consulta de cliente

```
----- Consulta de cliente -----  
  
Cedula del cliente: 703260035  
  
=== DATOS DEL CLIENTE ===  
Cedula : 703260035  
Nombre : David Calvo  
Telefono: 87334321  
  
=== PEDIDOS DEL CLIENTE ===  
Codigo    Fecha      Subtotal  Total      Cant. libros  
-----  
1004      10/09/2024  $100000.00 $113000.00 10  
1005      16/09/2025  $10000.00  $11300.00  2
```

En este menú, se solicita la cedula del cliente y posteriormente se muestran los datos del cliente como su cédula, nombre y teléfono, también sus pedidos.

3) Salir

Regresa al menú principal.

C) Salir

El programa finaliza.

Descripción del Problema

El proyecto consiste en el desarrollo de un sistema de gestión de inventario y pedidos para una librería, implementado en el lenguaje de programación C. El propósito principal es simular una situación real donde un negocio necesita administrar de forma organizada y eficiente la información de los productos que ofrece (libros), los clientes que compran y los pedidos que se realizan diariamente.

El problema que se busca resolver es que, si este proceso se lleva de manera manual, es muy fácil cometer errores: por ejemplo, vender un libro sin tener existencias, registrar dos veces al mismo cliente, o no tener un historial claro de los pedidos que ya se generaron. Además, sin un sistema que persista la información, es difícil consultar rápidamente datos importantes como: cuáles son los libros más vendidos, cuáles clientes compran con más frecuencia o cuánto ha vendido la librería en un periodo de tiempo.

Por esa razón, el sistema debe incluir varias funcionalidades: registrar y actualizar libros en inventario, registrar clientes, crear y consultar pedidos, así como generar estadísticas de ventas. Todo esto se debe manejar con archivos de texto para asegurar la persistencia de los datos, de manera que al volver a ejecutar el programa la información registrada siga estando disponible.

A nivel técnico, este proyecto permite aplicar de forma práctica muchos de los conceptos vistos en el curso, como el uso de punteros, memoria dinámica, estructuras (struct), modularidad, paso de parámetros y manejo de archivos. También se exige implementar un sistema de menús en consola, con opciones tanto para el administrador (con acceso mediante usuario y contraseña) como para consultas generales accesibles a cualquier usuario.

En resumen, el problema que aborda este proyecto es cómo diseñar e implementar, desde cero y en lenguaje C, un sistema que organice y administre de manera adecuada la información de una librería, evitando errores, mejorando la consulta de datos y permitiendo obtener estadísticas útiles para la gestión del negocio.

Diseño del Programa

A. Decisiones de diseño

- Persistencia simple en archivos de texto planos (.txt / .json).
 - La configuración se lee/escrbe con `cargarConfiguracion` y `guardarConfiguracion` desde/ hacia `config.json`.
- Estructuras de dominio en headers:
 - Libros: `struct Libro` en `libro.h`.
 - Clientes: `struct Cliente` en `cliente.h`.
 - Pedidos y detalles: `struct Pedido` y `DetallePedido` en `pedido.h`.
 - Registros para estadísticas: `Registro` en `registro.h`.
- Separación en capas/archivos:
 - Interfaz / navegación: `menu.c` + `menu.h` (entrada y menús).
 - Lógica de datos y persistencia: `datos.c` + `datos.h`.
 - Utilidades de strings/memoria: `auxiliares.c` + `auxiliares.h`.
 - Configuración: `config.c` + `config.h`.
- Gestión de memoria:
 - Uso explícito de `malloc`/`realloc`/`free`. Funciones auxiliares para clonar cadenas: `asignarString`. Comparación propia: `compararString`.

B. Algoritmos usados

- Lectura de archivos y parseo:
 - Lectura línea por línea en `leerArchivo` y separación por delimitador en `separarTexto`. Operan en tiempo lineal respecto a las líneas por archivo.
- Búsquedas:
 - Búsquedas por código/cedula se hacen por recorrido lineal (p. ej. `buscarLibroPorCodigo`, `buscarClientePorCedula`),
- Actualizaciones de arrays dinámicos:
 - Inserciones/elim. usan `realloc` y desplazamiento.
- Ordenamiento:
 - Utilizandose para clientes con más pedidos, `ordenarPorCantidadDesc` en `[programa/datos.c]` utiliza un bubble-sort.
- Estadísticas / agregaciones:
 - Construcción de agrupaciones (por año, por cliente, por libro, por autor) se realiza acumulando en arreglos dinámicos y buscando entradas existentes con búsqueda lineal.
- Persistencia:
 - Reescritura completa de archivos para operaciones como actualizar inventario o reescribir pedidos (Funciones: `actualizarTodosLibros`, `actualizarTodosClientes`, `reescribirArchivosPedidos`).

C. Diagrama de Archivos

- programa/main.c — entrada y arranque del menú.
- programa/menu.c / menu.h — interacción con el usuario y flujos.
- programa/datos.c / datos.h — carga, búsqueda, actualización de libros, clientes y pedidos; generación de estadísticas.
- programa/config.c / config.h — carga/guardado de config JSON y gestión de IDs.
- programa/auxiliares.c / auxiliares.h — utilidades para strings y memoria.
- Headers de tipos:
 - programa/libro.h (struct Libro)
 - programa/cliente.h (struct Cliente)
 - programa/pedido.h (struct Pedido y DetallePedido)
 - programa/registro.h (tipos para estadísticas)
- Archivos de datos:
 - programa/libros.txt, clientes.txt, pedidos.txt, detalles.txt, CARGAR.txt, config.json

Librerías Usadas

- **stdio.h** — manejo de archivos y entrada/salida (fopen, fclose, fprintf, fscanf, fgets, printf). Usado en: config.c, datos.c, menu.c, main.c.
- **stdlib.h** — gestión de memoria dinámica, conversiones y utilidades (malloc, realloc, free, atoi, exit). Usado en: auxiliares.c, datos.c, config.c, menu.c.
- **string.h** — operaciones con cadenas (strcpy, strcmp, strlen, strtok). Usado en: auxiliares.c, datos.c, config.c, menu.c.
- **stdbool.h** — tipo booleano (bool, true, false). Usado en: archivos de lógica (datos.c, menu.c).
- **time.h** (si hay manejo de fechas/tiempos) — formato/validación de fechas o timestamps (presente solo si se usa en registros/estadísticas).
- **Headers locales:** auxiliares.h, datos.h, menu.h, config.h, libro.h, cliente.h, pedido.h, registro.h — definen tipos y prototipos usados por los .c.

Análisis de resultados

Objetivo	Porcentaje de realización	Justificación
Registrar libros en inventario	100%	Se implementó correctamente el registro de libros con validación de código único y almacenamiento en estructuras dinámicas.
Actualizar inventario (carga en lote)	100%	Se desarrolló el módulo para modificar cantidades desde archivo externo, validando que no existan valores negativos.
Registrar clientes	100%	Se implementó el registro con validación de cédula única y validaciones de formato en los datos ingresados.
Crear pedidos	100%	El sistema permite agregar, eliminar y modificar líneas de pedido, validando el stock disponible y descontando al confirmar.
Consultar pedidos	100%	Se listan todos los pedidos generados y es posible ver el detalle completo de cada uno.
Generar estadísticas (ventas, clientes frecuentes, libros más vendidos)	100%	Se implementaron todas las estadísticas requeridas, incluyendo filtros por año.
Consulta de catálogo	100%	Se despliega correctamente la lista de libros, con opción de filtrar por autor.
Consulta de cliente	100%	Se muestran los datos del cliente y su historial de pedidos con totales.
Seguridad de acceso (usuario y contraseña administrador)	100%	El sistema restringe correctamente el acceso a las funciones administrativas mediante validación de credenciales.
Persistencia de datos en archivos de texto	100%	Toda la información se guarda y carga desde archivos de texto.

Objetivo	Porcentaje de realización	Justificación
Modularidad y uso de punteros, memoria dinámica, paso de parámetros	100%	El programa fue diseñado de forma modular, usando struct, punteros y memoria dinámica con control de malloc y free.
Documentación interna y externa	100%	Se incluyó documentación en el código y en el externo según lo solicitado.

Justificación de toma de decisiones

A. Punteros en secciones de código. Ejemplos concretos de paso por valor y por referencia

Punteros:

- Strings en C: las cadenas son `char*`; usar punteros es obligatorio para manejar nombres, códigos y textos.
- Datos dinámicos y vida útil: objetos creados en heap (libros, clientes, pedidos, configuración) deben sobrevivir al retorno de funciones, por eso se devuelven punteros; el llamador es responsable de liberar.
- Arrays dinámicos (`Libro* libros`, `Cliente* clientes`, `Pedido* pedidos`) requieren punteros para realloc, y compartir la misma región de memoria entre funciones.
- Eficiencia / evitar copias: structs pueden contener varios campos (cadenas, arrays). Pasar punteros evita copiar estructuras grandes y reduce coste.
- Funciones que modifican objetos (stock, secuencial, detalles del pedido) reciben punteros para cambiar el estado sin devolver copias.
- Relaciones y referencias: un Pedido puede referenciar Cliente o DetallePedido por puntero o por índice; punteros permiten modelar relaciones directas sin duplicar datos.
- Punteros permiten definir quién posee la memoria

Paso por valor y por referencia:

- Buscar y devolver una referencia en memoria:
 - Las funciones de búsqueda (por ejemplo, buscar un libro por código) devuelven un puntero al elemento dentro del array cuando lo encuentran. Esto evita copiar el libro y permite que se lea o modifique ese libro directamente. Si no existe, devuelven NULL para indicar ausencia.
- Modificación por referencia:
 - Para operaciones que cambian estado (por ejemplo, reducir o aumentar stock de un libro, agregar un detalle a un pedido) las funciones reciben punteros a las estructuras para aplicar el cambio sobre el objeto original en memoria sin tener que devolver una copia modificada.
- Gestión de listas dinámicas:
 - Al insertar un nuevo cliente o pedido se utiliza la combinación de un puntero al array (y en algunos casos puntero a puntero) junto con el tamaño, para llamar a realloc y ampliar la colección; la función modifica el array compartido y actualiza el contador, por eso opera sobre referencias y no sobre copias locales.

- Creación y retorno de estructuras al heap:
 - Las funciones que crean entidades (por ejemplo, crear un pedido nuevo) asignan la estructura en el heap y retornan un puntero. Esto garantiza que el objeto siga existiendo después de que la función retorne y define claramente quién debe liberar esa memoria.
- Paso por valor cuando no se modifica:
 - En operaciones meramente de mostrar y con estructuras pequeñas (por ejemplo, una función que imprime datos de un cliente sin modificarlos), puede usarse paso por valor para indicar que la función no altera el original y que una copia temporal es suficiente.

B. ¿Cómo se asegura la modularidad y el ocultamiento de la información?

- Separación por responsabilidad (cada .c tiene una capa):
 - menu.c: interfaz y flujo del usuario.
 - datos.c: lógica de dominio y persistencia.
 - config.c: carga/guardado de configuración.
 - auxiliares.c: utilidades comunes.
- Encapsulación mediante headers y funciones públicas
 - Solo los prototipos necesarios se exponen en los *.h (datos.h, menu.h, config.h, libro.h, cliente.h, pedido.h). El resto son funciones internas.
- Ocultamiento con funciones estáticas
 - Funciones internas que no deben exportarse (p. ej. extraerValorJSON en config.c) se declaran static para restringir su visibilidad.
- Minimizar acoplamiento
 - Tipos compartidos (structs en libro.h / cliente.h / pedido.h) permiten comunicación controlada; la lógica de persistencia está en datos.c, no dispersa por la UI.

Bitácora

Fecha	Usuario	Comentario
2025-08-30	johnsydev	Inicio de carpetas
2025-08-30	johnsydev	Inicio de menús
2025-08-30	Johnsydev	Lectura de archivos, separador de texto, autenticación de administradores
2025-08-31	Johnsydev	Lectura de archivos, separador de texto, autenticación de administradores
2025-09-01	Johnsydev	Registrar libro
2025-09-01	Johnsydev	Modificar inventario y stock, auxiliares
2025-09-01	Dcalvog25	Registrar Cliente Funcionalidad de opciones administrativas: Registrar Clientes.
2025-09-04	Dcalvog25	Creacion de Pedidos Se realizo la funcionalidad de crear pedidos, con el menú de gestión del mismo. Además, de otros cambios y mejoras de errores y warnings.

2025-09-04	Dcalvog25	Separacion Pedidos y Detalles Se realizo la separacion en dos archivos pedidos (encabezado) y detalles del pedido.
2025-09-07	Erincam09	Creacion de Estadisticas
2025-09-07	Erincam09	Cambios en los prints
2025-09-07	Erincam09	Consulta por Cliente y Consulta de Pedidos
2025-09-07	Erincam09	Funcionalidades
2025-09-13	johnsydev	Memoria dinámica en manejo de strings
2025-09-14	Johnsydev	Avances
2025-09-14	Dcalvog25	Valores iniciales en json y Eliminar Pedido
2025-09-14	Erincam09	Avances Finales

2025-09-15	Dcalvog25	Modificaciones en Generar Pedido
2025-09-15	Dcalvog25	Mejoras en errores
2025-09-16	Erincam09	Validacion de Stock
2025-09-16	Dcalvog25	Mejoras en ModificarPedido
2025-09-16	Johnsydev	Validaciones
2025-09-16	Dcalvog25	DocumentacionExt, Txt y Carpetas