



# Facultad de Matemáticas UADY

Propuesta de Proyecto

*Asignatura:*

Programación estructurada

*Docente:*

Emilio Gabriel Rejón Herrera

*Integrantes equipo:*

Campos Daguer Emilio

Couoh Martin Reynaldo

Graniel Arzat Aaron Isaac

Herrera Herrera Adiel Elioenai

Méndez Sierra Daniel

## Entrega final

### Mapeo de Requerimientos

Revisión de la funcionalidad de los requerimientos con base en el sistema desarrollado.

Requerimientos Funcionales		Implementado	Requerimientos No funcionales		Implementado
<b>RF001</b>	Identificación	Si	<b>RNF001</b>	Interfaz intuitiva	Si
<b>RF002</b>	Registrar Empleados	Si	<b>RNF002</b>	Inventario ordenado	Si
<b>RF003</b>	Editar registros de empleados	Si	<b>RNF003</b>	Interfaz visualmente cómoda	Si
<b>RF004</b>	Facturación	Si	<b>RNF004</b>	Ejecutarse varias veces	Si
<b>RF005</b>	Guardar información de venta	Si	<b>RNF005</b>	Opciones.	Si
<b>RF006</b>	Información de ventas anteriores	Si	<b>RNF006</b>	Sistemas operativos disponibles	Si
<b>RF007</b>	Visualizar inventario de la bodega.	Si	<b>RNF007</b>	Clave de empleado	Si
<b>RF008</b>	Visualizar inventario de la tienda.	Si	<b>RNF008</b>	Contraseña	Si
<b>RF009</b>	Asistente de caja	Si	<b>RNF009</b>	Código de producto	Si
<b>RF010</b>	Realizar corte de caja	Si	<b>RNF010</b>	Permitir acceso.	Si
<b>RF011</b>	Realizar Apertura de caja	Si	<b>RNF011</b>	Administrar registros de empleados.	Si
<b>RF012</b>	Realizar venta	Si	<b>RNF012</b>	Tiempo de respuesta.	Si
<b>RF013</b>	Devolución	Si	<b>RNF013</b>	Actualizar Inventarios	Si
<b>RF014</b>	Ingresar productos	Si			
<b>RF015</b>	Defecto de Fabrica	Si			

RF01 6	Validar ingreso	Si
RF01 7	Menú principal	Si
RF01 8	Salir del sistema	Si

## Documentación

El código fue documentado a forma de comentarios conforme a las funciones específicas de las líneas de código. Se adjuntan capturas del código (código también disponible en el repositorio y comprimido de la entrega final).

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <conio.h>
5
6 // Interfaz de usuario tentativa (11/05/22)
7
8 FILE *fd;
9
10 #define LONGITUD 80
11 #define TECLA_ENTER 13
12 #define TECLA_BACKSPACE 8
13 #define NUMERO_USUARIOS 3
14
15 int main()
16 {
17     //Variables para guardar e imprimir la informacion de la ventas y la facturacion
18     char informacionVentas[] = "Informacion de las ventas anteriores.txt", informacionFactura[] = "Factura.txt";
19     int c;
20
21     //Variables para realizar venta
22     int codProductos, masProductos;
23     float ventaMulti, preciUn, ventaInd;
24
25     // Variable para elegir de las opciones del menu; V. para la cantidad de productos siendo comprados; V. para usar en el switch de inventario
26     int opcionMenu1, opcionMenu2, opcionMenu3, opcionMenu4, cantProductos, inventarioSwitch, usuarioAsistente, ventasTotal, devTotal;
27
28     // Variables para el corte y apertura de caja
29     int apertura, billMil, billQuin, billDoci, billCien, billCinc, billVen, monDiez, monCinc, monDos, monUno;
30     float proDinero, posDinero;
31
32     // Variables para iniciar sesion
33     char loginUsuario[ LONGITUD + 1 ], loginClave[ LONGITUD + 1 ], caracter;
34
35     // Servira para decidir en un switch que opcion realizar respecto al registro de usuarios
36     int registroUsuarios;
37
38     // Arreglos de usuarios y sus respectivas claves
39     char usuarios[ NUMERO_USUARIOS ][ LONGITUD + 1 ] = {"0001", "0002", "0003"};
40     char claves[ NUMERO_USUARIOS ][ LONGITUD + 1 ] = {"12", "34", "56"};
41
42     int i = 0, k = 0, j = 0;
43
44     //Inicio de sesion- Reynaldo
45     printf("\n\t\t\t\t\tINICIO DE SESION\n");
46     do
47     {
48         // 'system("cls")' limpia la pantalla - Daniel
49         system("cls");
50         printf("\n\t\t\t\t\tUSUARIO: ");
51         gets(loginUsuario);
52         printf("\t\t\t\t\tCLAVE: ");
53
54         // Hace que la contrasenna ingresada no sea visible; es cubierta por '*' - Daniel
55         while (caracter = getch())
56         {
57             if (caracter == TECLA_ENTER)
58             {
59                 loginClave[i] = '\0';
60                 break;
61             }
62             else if (caracter == TECLA_BACKSPACE)
63             {
64                 if (i > 0)
65                 {
66                     i--;
67                     printf("\b\b");
68                 }
69             }
70             else
71             {

```

```

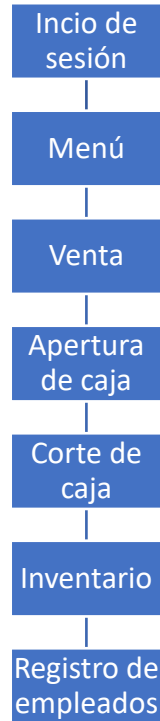
72         if (i < LONGITUD)
73         {
74             printf("***");
75             loginClave[i] = caracter;
76             i++;
77         }
78     }
79 }
80
81 //Se Valida al usuario - Reynaldo
82 for (j = 0; j < NUMERO_USUARIOS; j++)
83 {
84     if (strcmp( loginUsuario , usuarios[j] ) == 0 && strcmp( loginClave, claves[j] ) == 0)
85     {
86         k = 1;
87         break;
88     }
89 }
90 // Corregir lo del inicio de sesion (no se que sale mal, pero si lo escribes mal a la primera, ya no puedes iniciar sesion; se reinicia el bucle idk) :)
91 } while ( k == 0 );
92
93 while ( opcionMenu1 != 4)
94 {
95     // Las opciones son de ejemplo
96     printf("\n  1. Punto de venta");
97     printf("\n  2. Revisar inventario");
98     printf("\n  3. Administrar empleados");
99     printf("\n  4. Salir");
100
101     do
102     {
103         printf("\n Ingrese su eleccion (1-4):  ");
104         scanf("%d", &opcionMenu1);
105     } while ( opcionMenu1 < 1 || opcionMenu1 > 4);
106
107     switch ( opcionMenu1 )
108     {
109     case 1:
110         do
111         {
112             //Se despliega el menù principal - Daniel
113             printf("\n  1. Realizar venta");
114             printf("\n  2. Apertura de caja");
115             printf("\n  3. Corte de caja");
116             printf("\n  4. Revisar informacio de ventas anteriores");
117             printf("\n  5. Salir");
118
119             do
120             {
121                 printf("\n Ingrese su eleccion (1-5):  "); //Se le pide al usuario que ingrese una opcion
122                 scanf("%d", &opcionMenu2);
123             } while ( opcionMenu2 < 1 || opcionMenu2 > 5);
124
125             switch ( opcionMenu2 )
126             {
127             // Menu para realizar una venta -Emilio
128             case 1:
129                 if(apertura==1){
130                     ventaMulti=0;
131                     fd = fopen (informacionFactura, "wt");
132                     fprintf(fd, "\n\t\tFacturacion");
133                     fprintf(fd, "\n_____");
134                     fclose(fd);
135                     for(;;){
136                         printf("\n Ingrese el codigo que se va a comprar: ");
137                         scanf("%d", &codProductos);
138                         printf("\n Ingrese la cantidad de los productos: \n");

```

## Modularidad

## Organización general del sistema

El sistema se encuentra dividido en los siguientes módulos que se representan a través de funciones en el código:



Lo primero que pide el sistema es iniciar sesión, a continuación, se despliega el menú con las diversas funciones que el programa puede realizar. Cada módulo engloba tareas específicas:

**Venta:** esta parte incluye el cobrar un producto, imprimir un comprobante, factura; así como las devoluciones.

**Inventario:** permite editar el inventario a usuarios autorizados, así como consultarlo.

**Caja:** incluye las funciones de asistente de caja, abrir caja u corte de caja.

**Historial de ventas:** son las ventas registradas al igual que las mermas.

**Registrar empleados:** permite administrar a los trabajadores y registrarlos para el uso del programa.

## Organización del código con base en entradas, procesamiento y salidas

### Inicio de sesión

**Entrada:** Clave de empleado, clave de administrador o clave del administrador de base de datos.

**Proceso:** Validación de qué tipo de clave se ha introducido.

**Salida:** Menú.

## **Menú**

**Entrada:** Número de opción.

**Proceso:** Validar el número que se ha introducido.

**Salida:** Función de acuerdo con la opción elegida.

## **Cobrar un producto**

**Entrada:** Código del producto, forma de pago, cantidad. Si se requiere factura y si se quiere imprimir el comprobante.

**Proceso:** Validar los datos introducidos y guardarlos en la base de datos.

**Salida:** total, Ticket de compra y/o factura en caso de ser solicitado

## **Devoluciones**

**Entrada:** Código del producto, motivo de devolución.

**Proceso:** Guardar la información en la base de datos.

**Salida:** Registro de devolución.

## **Inventario**

**Entrada:** Opción a elegir:

- Editar inventario (sólo disponible para administrador de base de datos)
- Consultar inventario.

**Proceso:** Validar opción elegida. En caso de la segunda se volverá a validar si se quiere consultar el inventario de tienda o de la bodega.

**Salida:** Acceso a la edición del inventario o a la información de este.

## **Caja**

**Entrada:** opción a elegir entre:

- Abrir caja
- Corte de caja:

**Proceso:** validar la opción seleccionada

- Abrir caja: se inicia el asistente de caja para ingresar el número de billetes y monedas de cada tipo
- Corte de caja: se confirma que se quiere definir el corte como cerrado

**Salida:** según sea el caso:

- Abrir caja: cantidad de dinero actual

- Corte de caja: cantidad de dinero final

### **Historial de ventas**

**Entrada:** se ingresa el día del que se quiere revisar el historial

**Proceso:** se busca la información en la base de datos

**Salida:** se muestra el historial

### **Administrar empleados (sólo disponible para administrador)**

**Entrada:** opción a elegir:

- registrar empleados
- editar registro de empleados
- eliminar registro de empleados

**Proceso:** según sea el caso:

- registrar empleados: se ingresa el nombre, clave y contraseña del nuevo empleado.
- editar registro de empleados: se introduce la clave de usuario que se desee editar y su nueva contraseña
- eliminar registro de empleados: se introduce la clave de usuario que se desee eliminar

**Salida:** registro actualizado.

### **Mapeo de los requerimientos con las funciones del sistema.**

<b>Requerimientos</b>		<b>Funciones del sistema</b>
<b>RF001</b>	Identificación	Validar entrada y solo dar acceso a los usuarios que estén registrados.
<b>RF002</b>	Registrar Empleados	Administrar registros de empleados.
<b>RF003</b>	Editar registros de empleados	Administrar registros de empleados.
<b>RF004</b>	Facturación	Después de cada compra se genera una factura con la información de la venta.
<b>RF005</b>	Guardar información de venta	Después de cada venta, dicha información se guardará para que esta pueda ser revisada en el futuro.
<b>RF006</b>	Información de ventas anteriores	Acceder a la información de ventas que se hayan hecho antes.

<b>RF007</b>	Visualizar inventario de la bodega.	Accede al inventario de la bodega.
<b>RF008</b>	Visualizar inventario de la tienda.	Accede al inventario de la tienda.
<b>RF009</b>	Asistente de caja	Despliega un menú que ayuda a sumar el dinero que se tenga en caja.
<b>RF010</b>	Realizar corte de caja	Realiza un corte de caja y no permite que se ingrese alguna otra compra, sin antes no hacer una apertura.
<b>RF011</b>	Realizar Apertura de caja	Realiza una apertura de caja permite que se ingresen las ventas del día.
<b>RF012</b>	Realizar venta	Ingresar un producto que se vaya a vender y este se eliminara del inventario automáticamente.
<b>RF013</b>	Devolución	Ingresar productos que hayan sido devueltos, si es por defecto, este será eliminado.
<b>RF014</b>	Ingresar productos	Ingresar productos que hayan llegado a la tienda.
<b>RF015</b>	Defecto de Fabrica	Dar salida a productos que estén defectuosos.
<b>RF016</b>	Validar ingreso	Validar entrada y solo dar acceso a los usuarios que estén registrados.
<b>RF017</b>	Menú principal	Menú que tiene como opciones: Revisar inventario, Punto de venta, Administrar empleados y Salir, si eres administrador, o Revisar inventario, Punto de venta y Salir, si eres empleado.
<b>RF018</b>	Salir del sistema	Cerrar el sistema.
<b>RNF001</b>	Interfaz intuitiva	Las instrucciones son claras y no tienen fallos ortográficos que dificulte su entendimiento.
<b>RNF002</b>	BDD ordenada	Tener ordenada la base de datos.
<b>RNF003</b>	Interfaz visualmente cómoda	Las instrucciones y los mensajes que se impriman son cortos, es decir, no contienen demasiada información.
<b>RNF004</b>	Ejecutarse varias veces	Tener abierto varias ventanas del mismo sistema, pero con diferentes opciones.
<b>RNF005</b>	Opciones.	Solo permite realizar una opción a la vez.
<b>RNF006</b>	Sistemas operativos disponibles	Disponibilidad en los sistemas operativos.
<b>RNF007</b>	Clave de empleado	Solo permite caracteres numéricos.
<b>RNF008</b>	Contraseña	Solo permite caracteres alfabéticos, numéricos y especiales.



<b>RNF009</b>	Código de producto	Solo permite caracteres alfabéticos y numéricos.
<b>RNF010</b>	Permitir acceso.	Solo permite el acceso a usuarios que tengan una clave de usuario y contraseña.
<b>RNF011</b>	Administrar registros de empleados.	Permite agregar o eliminar registros de empleados.
<b>RNF012</b>	Tiempo de respuesta.	El tiempo respuesta máxima del sistema es 5 segundos.
<b>RNF013</b>	Actualizar base de datos.	La base de datos se ira actualizando, dependiendo de las acciones que se realicen en el sistema.

## Proceso de desarrollo

### Reporte de participación individual en la codificación del proyecto

Durante la elaboración del código el equipo de trabajo realizó la documentación correspondiente a lo que realizó cada miembro del equipo:

Nombre	Tarea en el código
Daniel (Encargado)	-Dirigir el proceso de codificación -Interfaces de usuario -Documentación
Adiel	-Encargado de ligar el Excel a la base de datos -Documentación
Reynaldo	-Validación de usuarios -Creación de Usuario -Documentación
Aaron	-Menú -Documentación
Emilio	-Asistente de caja -Documentación

Comprimido y código disponible en: <https://github.com/Emcamdag/Sistema-de-cobro-e-Inventarios-Insomnes>

### Contribución general basado en elementos objetivos y medibles, de la participación individual en todo el proyecto.

Se retomaron las principales herramientas de distribución y métrica de contribución del trabajo individual en el equipo:

## Bitácora

En la bitácora se documentó quienes serían los principales encargados de cada sección a entregar de cada parte del proyecto y demás actividades a realizar:

### Viernes 13 de mayo de 2022-

Se dividió el trabajo en partes iguales para poder terminar el proyecto en el tiempo restante:

Aaron: Encargado de la parte de la revisión de la funcionalidad de los requerimientos con base en el sistema de desarrollo y código.

Adiel: Organización general del sistema, métrica de contribución y código.

Daniel: Mapeo de los requerimientos, organización del código y código

Reynaldo: Cartel científico y organización del código y código

Emilio Administración de la subida de documentos, codificación del repositorio y revisión del código.

## Métrica de contribución:

En esta se asignaron valores a las actividades a realizar para llevar un conteo de los porcentajes de contribución de cada miembro del equipo:

Métrica de contribución individual y de equipo - Entrega Final				
Ubicación de objetivo	Objetivo	Carga	Cumplimiento	Encargado
Mapeo de Requerimientos	Revisión de la funcionalidad de los requerimientos con base en el sistema de desarrollo	2	Si	Aaron
Documentación	Código fuente documentado	N/A	Si	TODOS
Modularidad	Organización general del sistema	1	Si	Adiel
	Organización del código en base en entradas, procesamiento y salidas	1 pt c/u	Si	Daniel y Reynaldo
	Mapeo de los requerimientos con las funciones del sistema	1	Si	Daniel
Proceso de desarrollo	Reporte de participación individual	N/A	Si	Todos
	Métrica de contribución	1	Si	Adiel
Presentación	Cartel científico	1	Si	Reynaldo
Extras indispensables	Administración de la subida de documentos y codificación del repositorio	2	Si	Emilio
	Código fuente bien documentado	N/A	Si	TODOS
	Total	10		
Contribución por Integrante	20% del 100% de actividades			

## Gráficas de las adiciones al repositorio en GitHub

Aquí se presentan los commits que cada integrante del equipo agregó al repositorio. La diferencia que se observa es por la subida de los contenidos del reporte al repositorio.



\*\*La cuenta de "TanoshiiSound" corresponde al miembro de equipo Daniel Méndez.

## Presentación

Se elaboró un cartel científico con la información relevante del proyecto la cual incluyó: Los objetivos, la descripción del producto de software, la organización del sistema, el diagrama a bloques, los casos de uso y la métrica de contribución de la última entrega.



También disponible en el siguiente enlace:

[https://www.canva.com/design/DAFAzz69ZZw/tbETnFy8roJ4blh3ztmDHg/view?utm\\_content=DAFAzz69ZZw&utm\\_campaign=designshare&utm\\_medium=link2&utm\\_source=sharebutton](https://www.canva.com/design/DAFAzz69ZZw/tbETnFy8roJ4blh3ztmDHg/view?utm_content=DAFAzz69ZZw&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)