

***Avances del Proyecto***

***NetCode***

**Profesor: Emilio Gabriel Rejón Herrera**

**Programación Estructurada**

**LIS – MEFI**

**Elaboró:**

Barón Pat Alan Josue

Cauich Davalos Victor Enrique

**Facultad de Matemáticas**

**Mapeo de Requerimientos**

**1.- Descripción exhaustiva de los requerimientos funcionales y no funcionales**

**Requerimientos Funcionales**

**RF\_01:** Cada vez que un administrador o empleado quiera ingresar, debe validarse por codificación el usuario y contraseña de los mismos, en caso de no existir una cuenta se deberán crear a través del administrador.

**RF\_02:** Al finalizar un pedido, siempre se deberá de realizar un ticket, en el que estén señalados los productos solicitados, así como la cantidad de los mismos, también se deberá señalar el costo individual de cada producto y el costo total que es el que se deberá pagar. Al igual que se debe de guardar cada ticket en un archivo txt.

**RF\_03:** En cualquier momento un administrador puede cambiar precios tanto a productos y a ingredientes que se estén vendiendo en el establecimiento al igual de que cada precio de los productos estará guardado en un archivo diferente de txt.

**RF\_04:** Se podrán ver los tickets que ya se hayan creado, esto eligiéndolo en un interfaz intuitivo, en el cuál se pueden consultar los tickets por día, semana o mes respecto al archivo de txt correspondiente siempre y cuando se haya concluido el periodo laboral según el tipo de ticket a consultar.

**RF\_05:** Las ganancias diarias, semanales y mensuales se irán sumando al momento de que se genere cada ticket después de una venta, cada ganancia se irá guardando en archivos txt’s respecto al tipo de ganancia que se este generando al momento, para diferenciarlos se guardaran por día y una vez concluida la semana se guardarán todos los tickets generados por día para que se elabore el de las ganancias semanales, siguiendo el mismo procedimiento para las ganancias mensuales.

**RF\_06:** Las cuentas de los empleados solo podrán acceder al sistema en su horario de trabajo que ha sido prestablecido por el administrador. Esto quiere decir que es un tipo de seguridad para que no puedan acceder en el sistema en cualquier momento.

**Requerimientos No Funcionales**

**RNF\_01:** El sistema deberá responder de la forma más rápida posible al momento de que se solicite una acción a través de sus interfaces para evitar posibles retrasos en la atención hacia los clientes.

**RNF\_02:** El usuario debe validar su contraseña y el nombre con el que se registró, si no se está registrado, que el administrador debe registrarlo, este no podrá acceder al sistema a realizar actividades. Al igual que cada empleado tendrá un horario en específico de trabajo, esto almacenado por el sistema, en la cual no podrán ingresar en cualquier horario de trabajo.

**RNF\_03:** El sistema debe tener un interfaz en el cual el trabajador o administrador deba poder saber qué hace cada apartado y el cuál se le facilite el poder usar el sistema evitando cualquier posible malinterpretación del mismo que conlleve a un mal uso del software.

**RNF\_04:** El sistema está disponible en cualquier momento, aunque la empresa debe tener la conciencia de solo usarlo en el tiempo laboral a menos que se registre un nuevo trabajador o realizar un cambio de precio.

**2.- Elaborar los casos de uso para cada funcionalidad del sistema, donde se describa el proceso y las rutas de acción alternativas**

**Casos de uso:**

**1.- Login / Registro:** El usuario proporciona sus datos de sesión como el usuario y contraseña y el programa determinará si los datos de la cuenta están asociados a una cuenta del tipo administrador o el de un empleado.

1.1.- En caso de que alguno de los datos proporcionados este erróneo, el sistema desplegará una alerta de que los datos son incorrectos y se pedirá que los reingrese nuevamente.

1.2.- En el caso de que no exista una cuenta en el sistema le pedirá al usuario que cree una nueva cuenta, que al ser la primera cuenta del sistema esta la tomará como una cuenta de tipo administrador.

**2.- Cambio de Precios:** Una vez confirmado que el administrador haya iniciado la sesión, se le permitirá cambiar los precios de los productos e ingredientes.

2.1.- En el caso de que el administrador por error cambie los precios de algún producto o ingrediente, durante cada cambio aparece una ventana de confirmación.

2.2.-Si el administrador decide no realizar ningún cambio simplemente presiona en el botón de regreso.

**3.- Venta de Productos:** Las cuentas de administradores y empleados tienen la capacidad de realizar las respectivas transacciones que sus clientes requieran y los tickets generados se almacenan para el cálculo de las ganancias.

3.1.- En el caso de que el cliente no decida agregar o quitar algún ingrediente se pasará inmediatamente a la generación de su ticket de venta y los datos se almacenan.

3.2.- Dado el evento de que no existan suficientes ingredientes o el inventario este vacío se emitirá un cuadro de alerta informando la insuficiencia del mismo.

**4.- Visualización de las Ganancias:** Cuando el usuario administrador inicie sesión este podrá consultar las ganancias que se general al día, semana o por mes.

4.1.- En el caso de las ganancias por día sólo se podrá consultar después de que finalice el horario laboral.

4.2.- Respecto a las ganancias por día sólo se podrá consultar una vez finalizada la semana o el de la semana anterior.

4.3.- Las ganancias mensuales se visualizarán una vez concluido el mes ó puede consultar la del mes pasado.

**5.- Visualización de los tickets:** Las cuentas de administradores pueden consultar los tickets generados por el sistema para la corroboración de las ganancias que se generan.

5.1.- En el caso de la consulta de los tickets por día, semana o mes debe haberse concluido dicho periodo laboral, de caso contrario se consultarán tickets previos.

**3.- Mapeo de los requerimientos (matriz de requerimientos), con el código correspondiente.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Matriz de requerimientos** | | | |
| **Requerimientos Funcionales** | | | |
| **Código** | **Descripción breve** | **Prioridad** | **Objetivos** |
| RF\_01 | Validación del usuario | Alta | 1 |
| RF\_02 | Creación de los tickets | Media | 3 |
| RF\_03 | Cambio de precios | Alta | 7 |
| RF\_04 | Consulta de tickets | Baja | 3 |
| RF\_05 | Ganancias registradas | Alta | 3,4,5 y 6 |
| RF\_06 | Horario laboral | Media | 1 |
| **Requerimientos No Funcionales** | | | |
| RNF\_01 | Sensibilidad del sistema | Alta | 3 |
| RNF\_02 | Seguridad | Alta | 1 y 7 |
| RNF\_03 | Interfaz intuitiva | Media | 1,2,3 y 7 |
| RNF\_04 | Accesibilidad del software | Media | 1 |

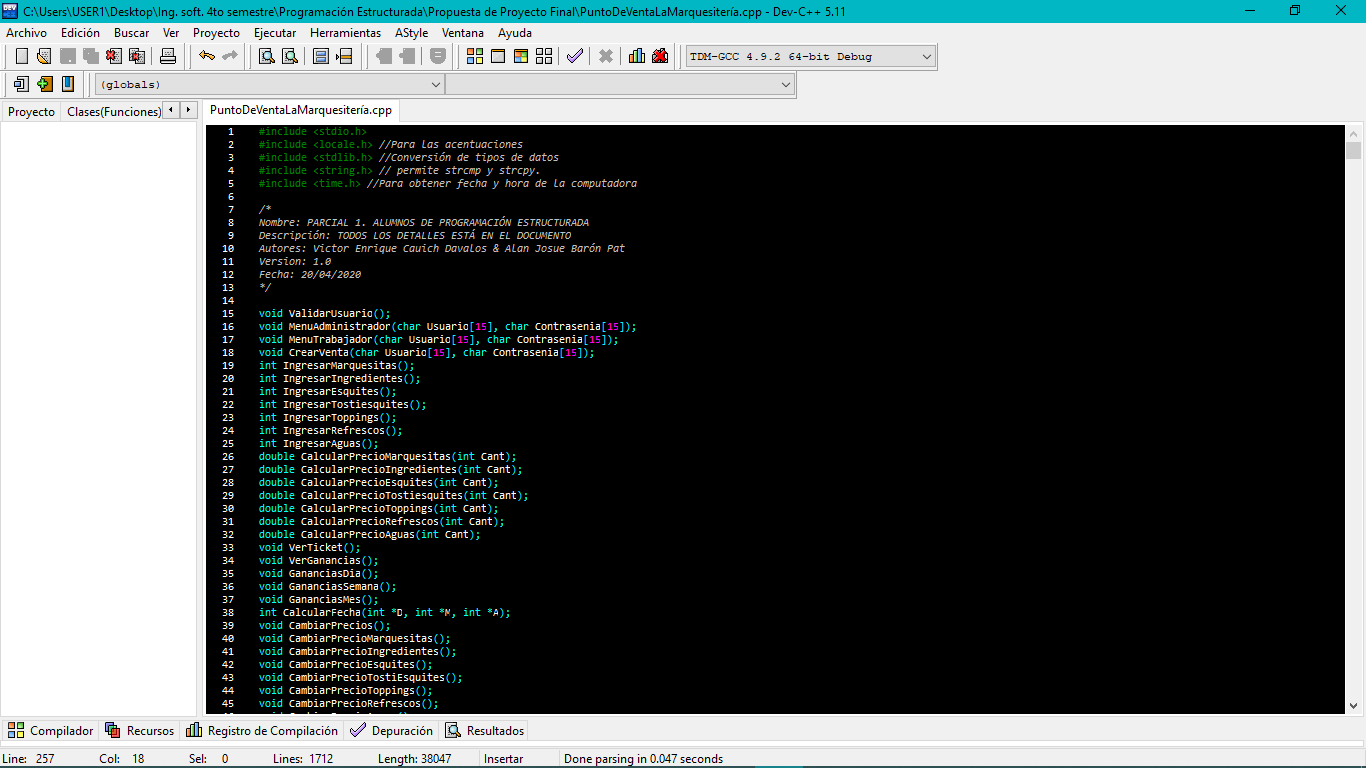
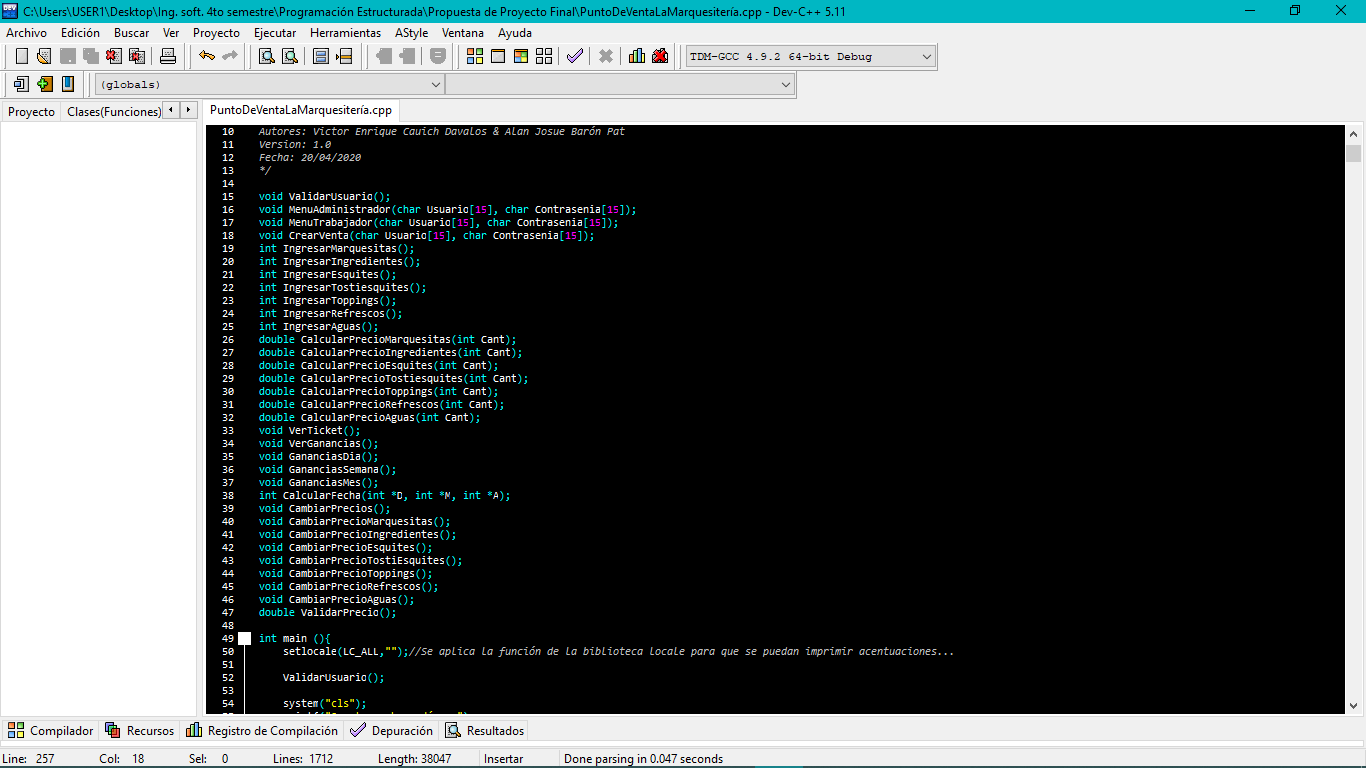
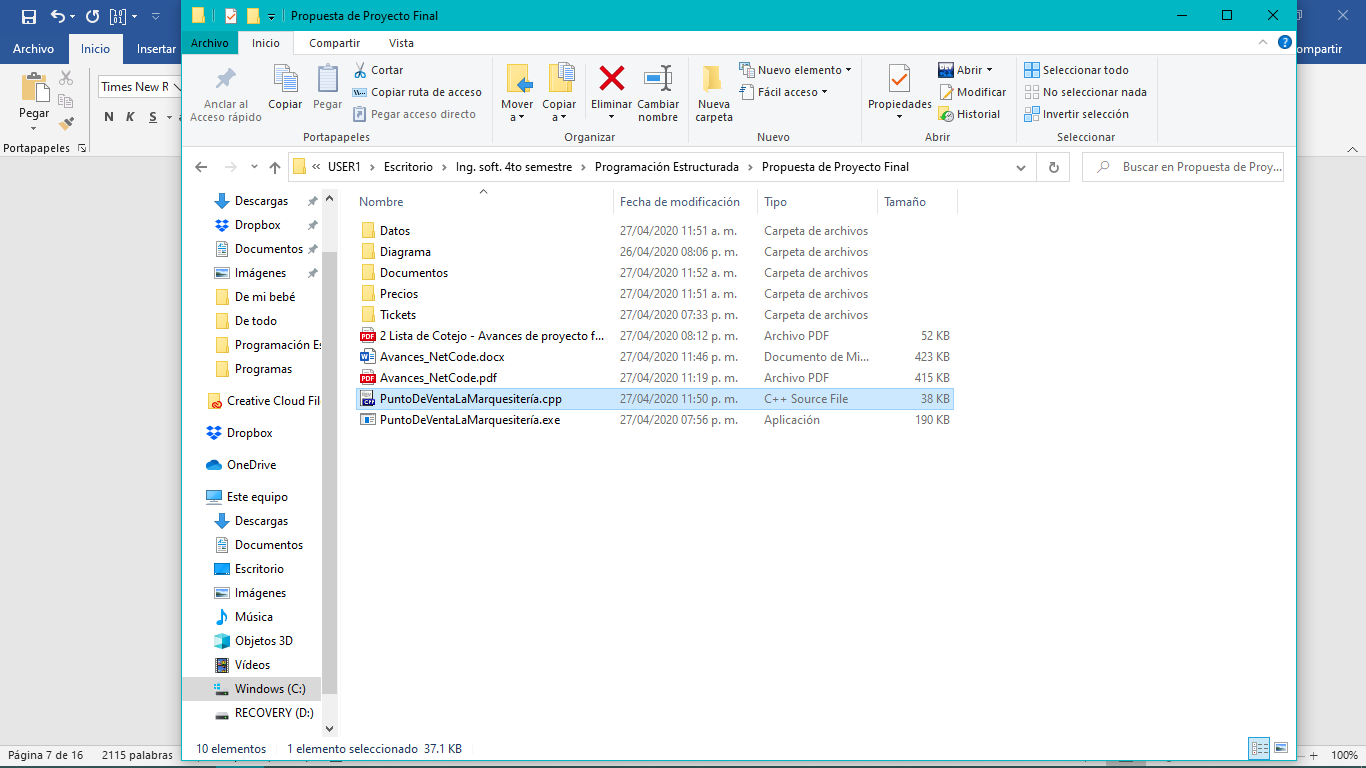
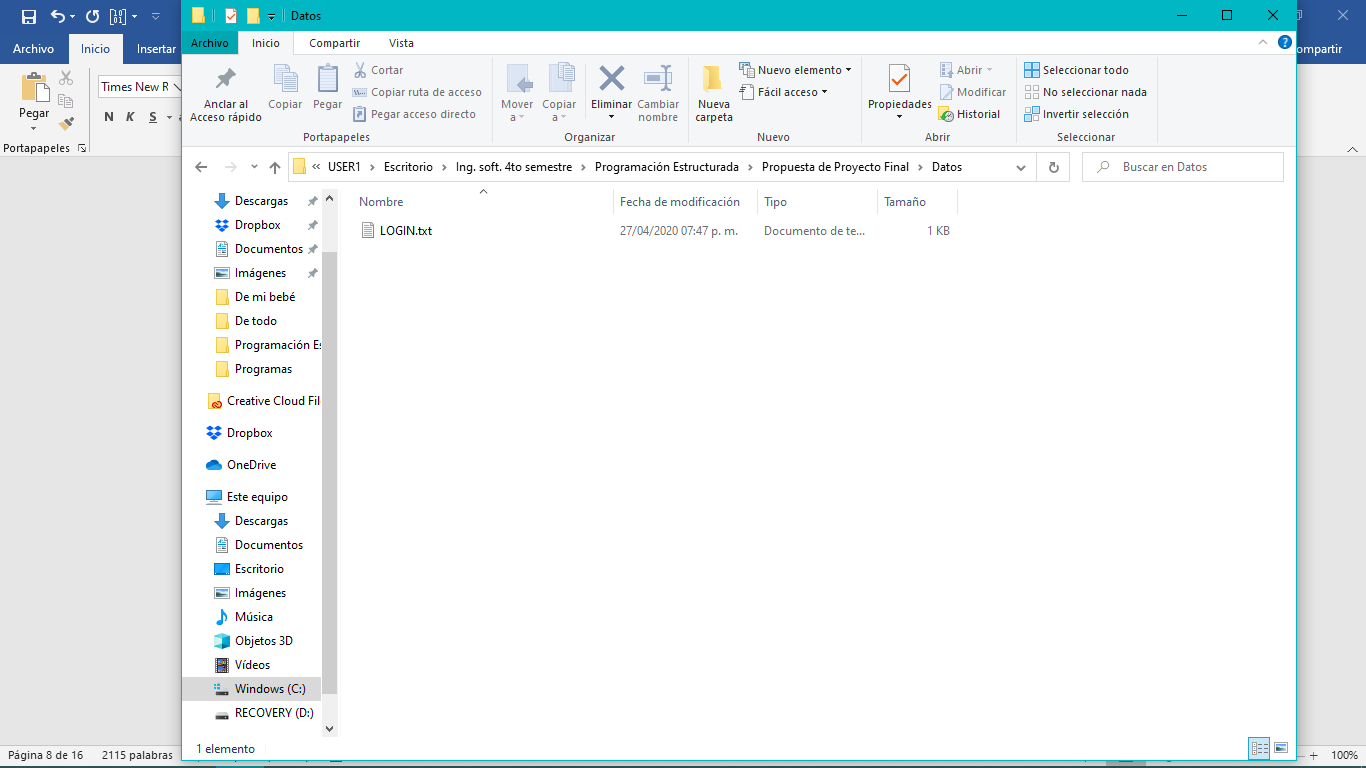
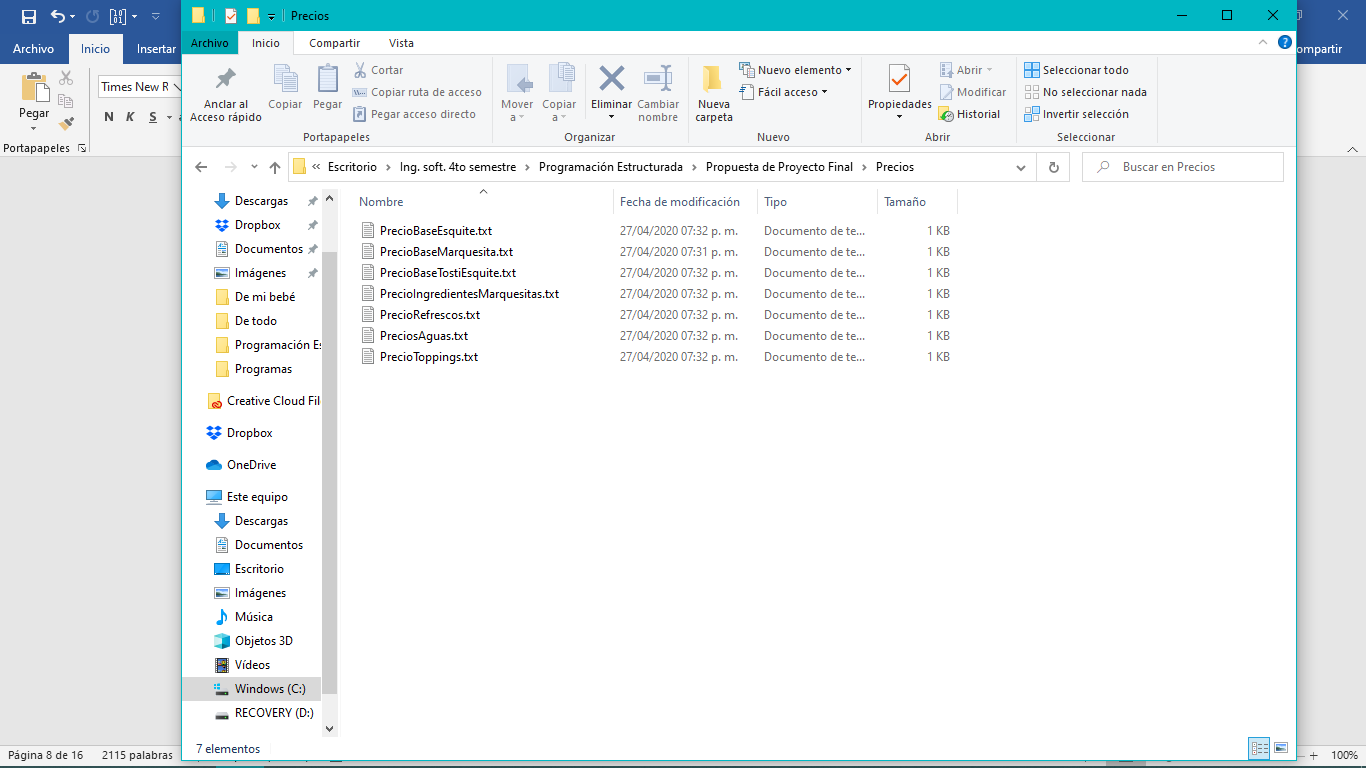
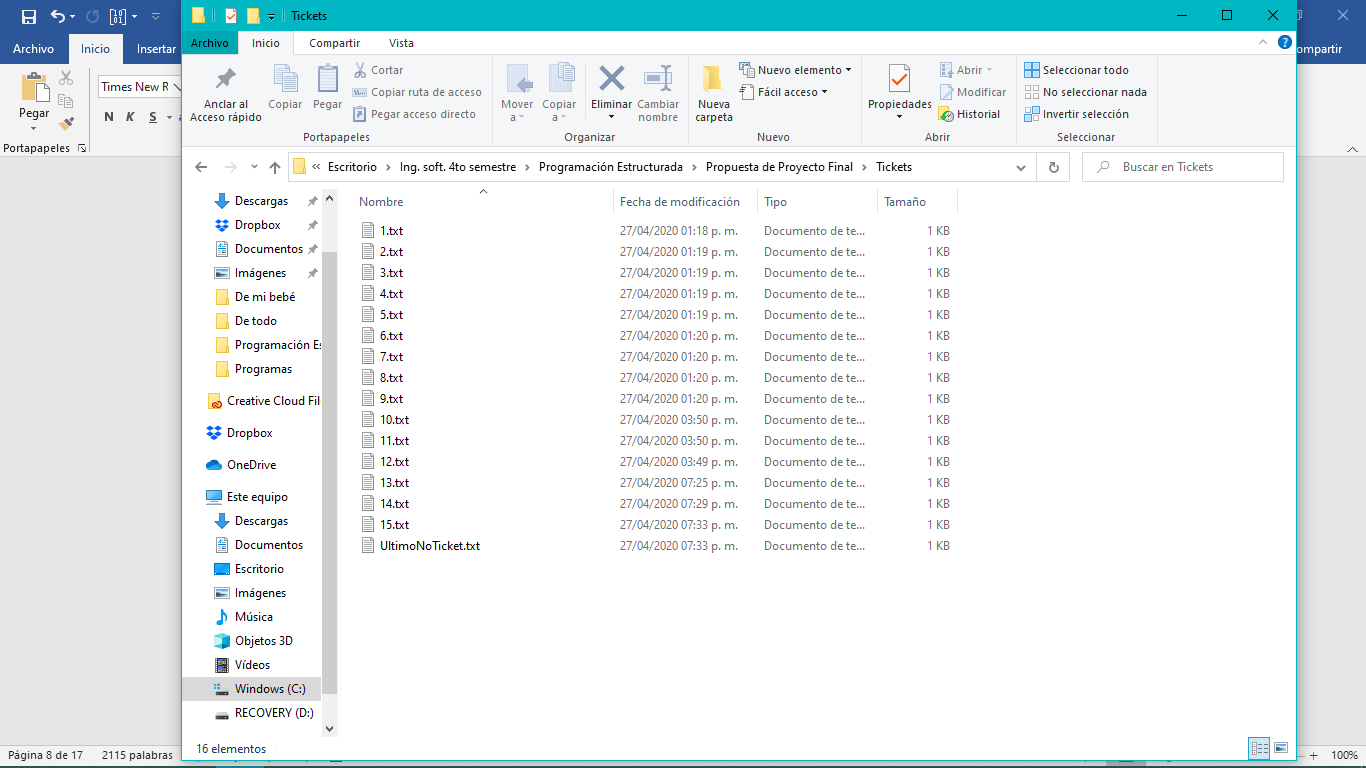
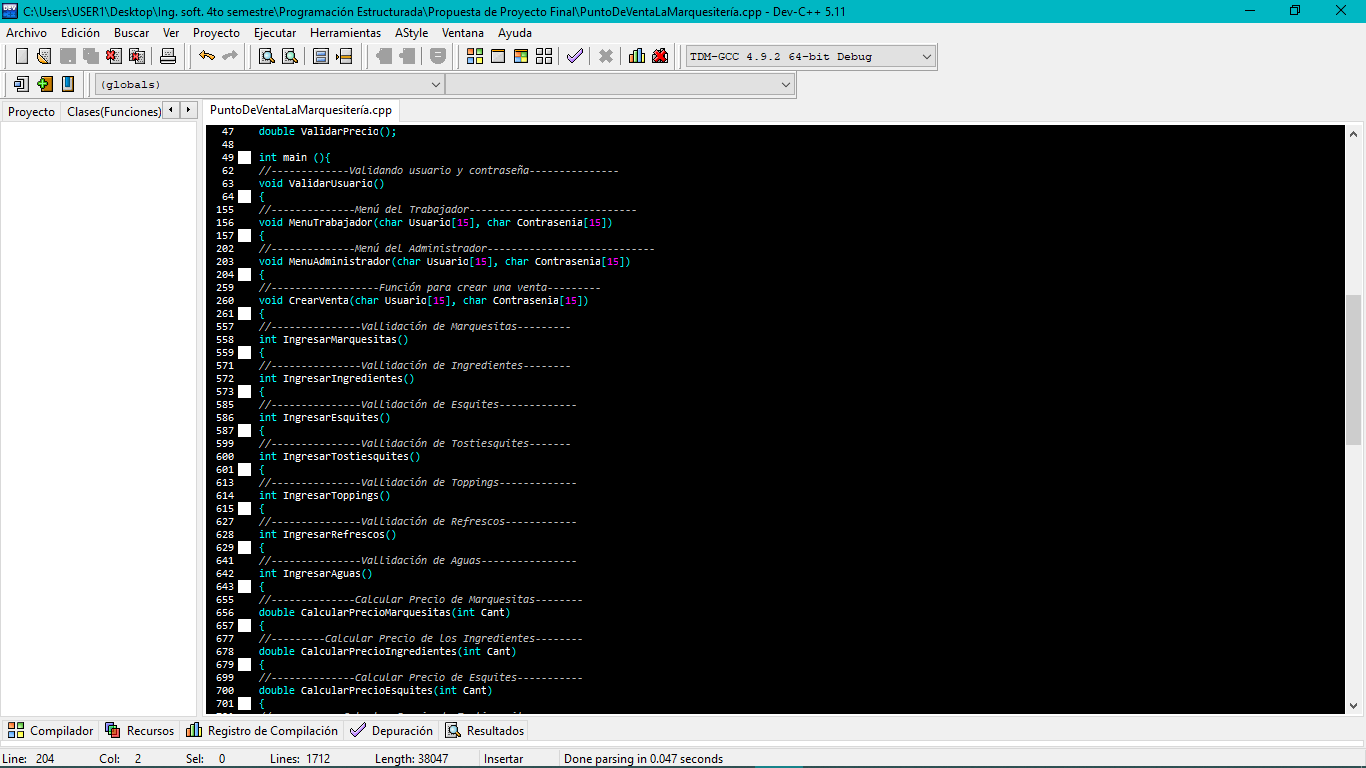
**4.- Informe del avance general de la implementación de dichos requerimientos.**

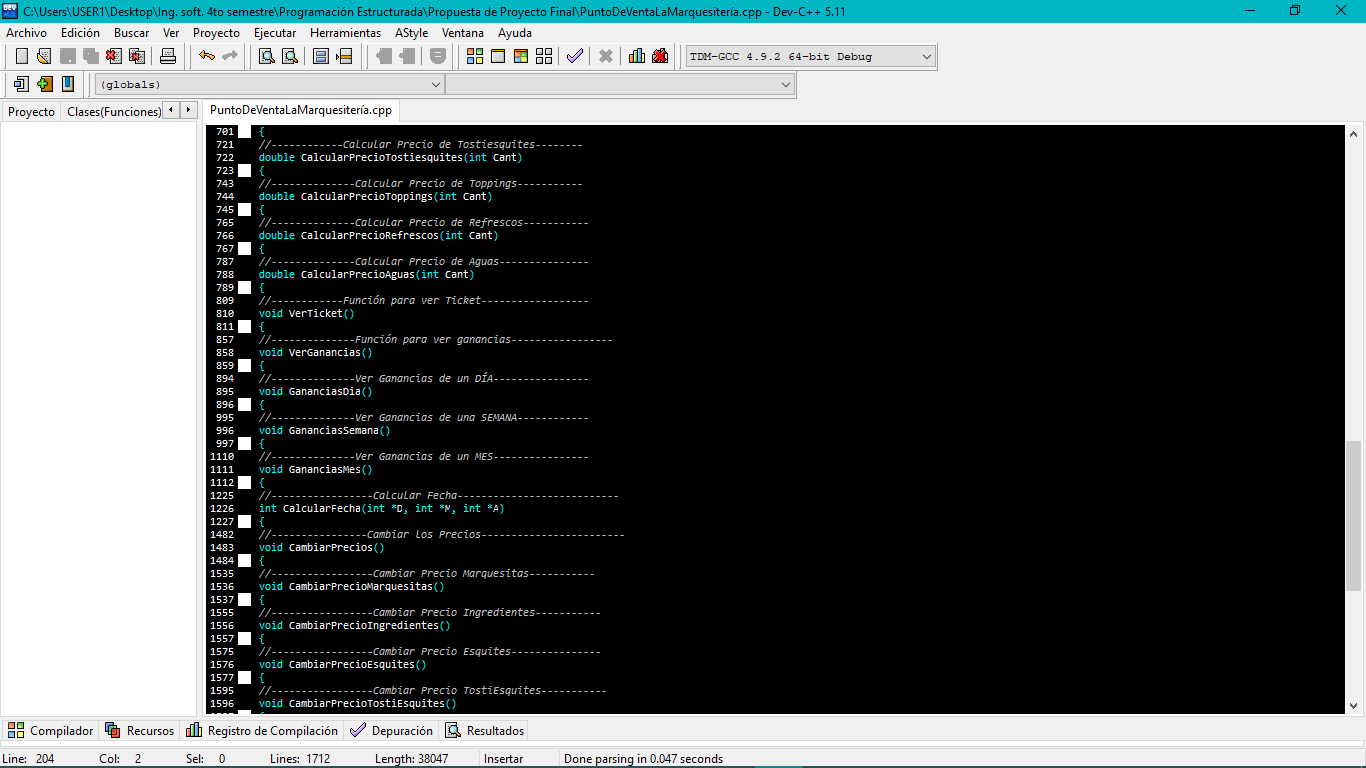
De acuerdo a lo que ya tenemos, ya está básicamente el 80% del software o proyecto, ya se implementaron varias funciones en la cual nos facilita la modularidad de nuestro proyecto, al igual que a un futuro se podría actualizar el software mas fácilmente, como algunos ajustes y mejoras de lo ya realizado que queremos realizar.

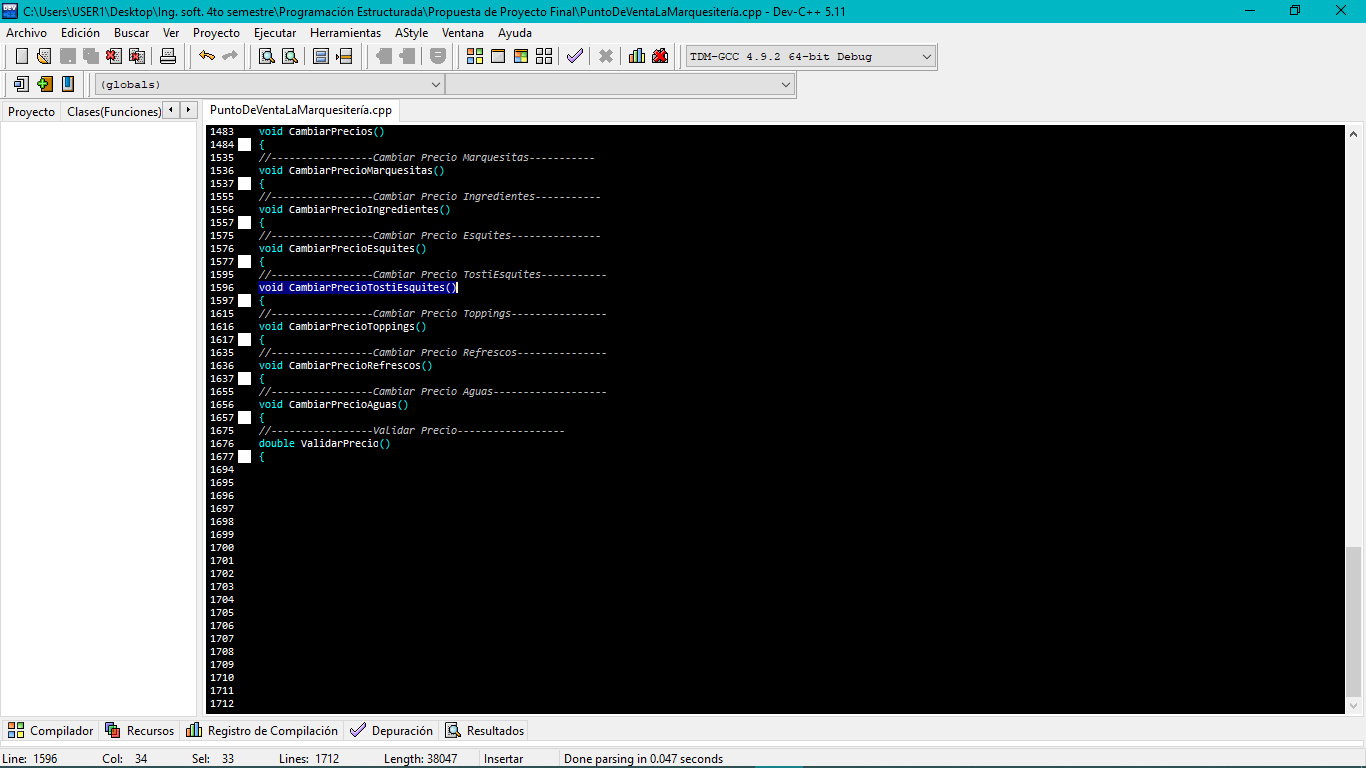
Por otro lado, los requerimientos en los cuales ya avanzamos son:

1. RF\_01. Validación de usuario: en la cual ya está implementado para saber quien es Administrador al igual que para saber quién es un simple trabajador, también se valida que estén correctamente los usuarios y contraseñas, esto para saber con serteridad quien pertenece a la empresa y quién no. Solo faltaría si un empleado está intentando entrar en su horario laboral o no.
2. RF\_02. Creación de los Tickets: Esto ya esta debidamente incorporado al sistema, quiere decir que no encontramos ninguna falla respecto a la creación de los mismos, inclusive se le agregan horarios del sistema, es decir de dispositivo u ordenador. Cabe recalcar que incluye validaciones de agregar productos, esto quiere decir que no se podrá crear un ticket vacío o que de igual forma se puedan agregar ingredientes cuando no se tiene agregado ninguna marquesita al pedido.
3. RF\_03. Cambio de Precios: Ya esta debidamente agregado, inclusive se realiza una validación en caso de que intenten cambiar el precio a algo negativo, esto quiere decir que la empresa no tendrá errores al momento de generar sus ganancias o cobros.
4. RF\_04. Consulta de Tickets: ya está debidamente implementado igual, esto quiere decir que tanto verificar por día, por semana o en un mes (28 días) las ganancias, de igual forma tiene validación, por si se trata de buscar una fecha que no exista, esto se imprimirá como que no se encontraron las ganancias indicadas; Cabe mencionar que se puso entre paréntesis que un mes se contempló como 28 días, debido a que la empresa “La Marquesitería”, hace un corte cada 4 semanas, esto para ver si se llegó a las ganancias para la renta, por ello se decidió manejar de esa forma.
5. RF\_05. Ganancias Registradas: esto se decidió realizar al instante que se solicité buscar las ganancias, y verificarlo por cada ticket el precio, esto se ahorra la creación de tickets. Es decir, que ya está debidamente implementado.
6. RF\_06. Horario Laboral: falta implementar una validación que el sistema permita y no permita por medio de esta el trabajo a empleados, esto quiere decir que si intenta acceder a un horario que no esté guardado en el sistema en el que él laboral, no podrá acceder al sistema.
7. RNF\_01. Sensibilidad del sistema: el software tiene una alta sensibilidad, esto quiere decir que, al momento de solicitar una operación, se genera en cuestión de milisegundos, se realizó lo más optimo posible.
8. RNF\_02. Seguridad: Esta un 80% listo, debido a que como se ha mencionado con anterioridad, falta validar a cada usuario que no intente acceder en un horario en el que esa persona no trabaje.
9. RNF\_03. Interfaz Intuitiva: la interfaz es muy intuitiva, para que cualquier persona pueda entenderlo, se hizo de esta manera para que al momento de cobrar no se demore mucho la atención al cliente. Es decir, que ya está debidamente implementado.
10. RNF\_04. Accesibilidad del software: ya está debidamente implementado, mas que nada para los administradores, para que no cualquier persona pueda acceder al sistema, y no cualquier persona pueda realizar cambios.

**Definición del estándar de codificación**

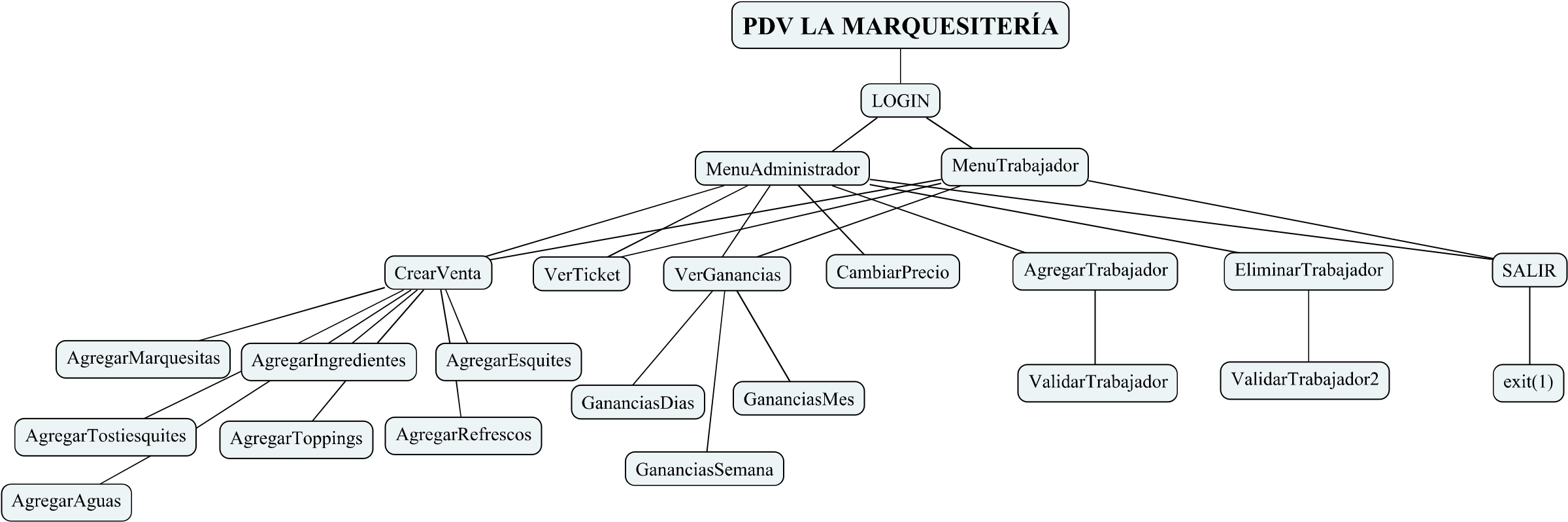
* El nombre de los identificadores utilizados se nombró de forma representativa para evitar errores de confusión y que indiquen que acciones están realizando, algunos de estos son:
* Usuario
* Contrasenia
* Val
* Val1
* Token
* Login
* Archivo
* Ultimo
* Aux
* Seleccion
* CantMarquesitas
* CantIngredientes
* CantEsquites
* CantTostiesquites
* CantToppings
* CantRefrescos
* CantAguas
* Etc.
* Las bibliotecas utilizadas en el proyecto fueron:
* No se esta realizando el uso de macros debido a que los datos se están guardando en archivos.
* Se están utilizando funciones por paso de parámetro o referencia, en donde se llevan a cabo los procesos del sistema como sería en la venta de productos. Algunos de estos, son:
* Los archivos son de uso esencial en el sistema pues se usan para guardar precios, los tickets, ganancias diarias, semanales y mensuales, al igual que los usuarios y contraseñas. Inclusive se dividen por carpetas, como verán a continuación:
* Los comentarios se hacen dentro del código con el uso de la doble barra (//) indicando que función realiza cada parte del código. Por ejemplos, antes de empezar un código, se escribe antes para qué va a servir, como se muestra a continuación:





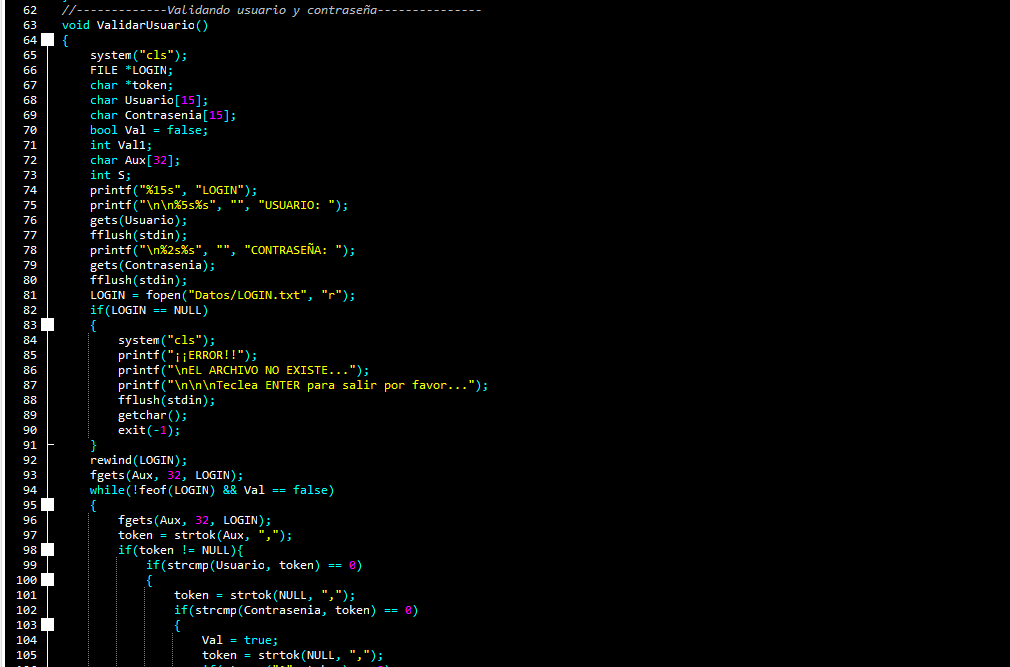
**Modularidad**

**1.- Organización del sistema en general**

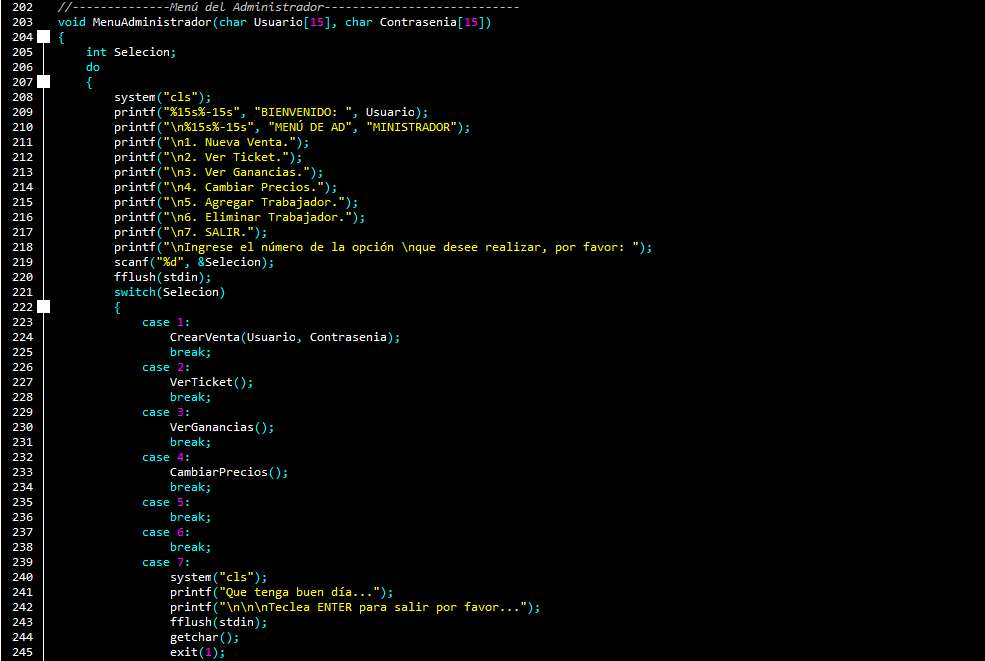
****

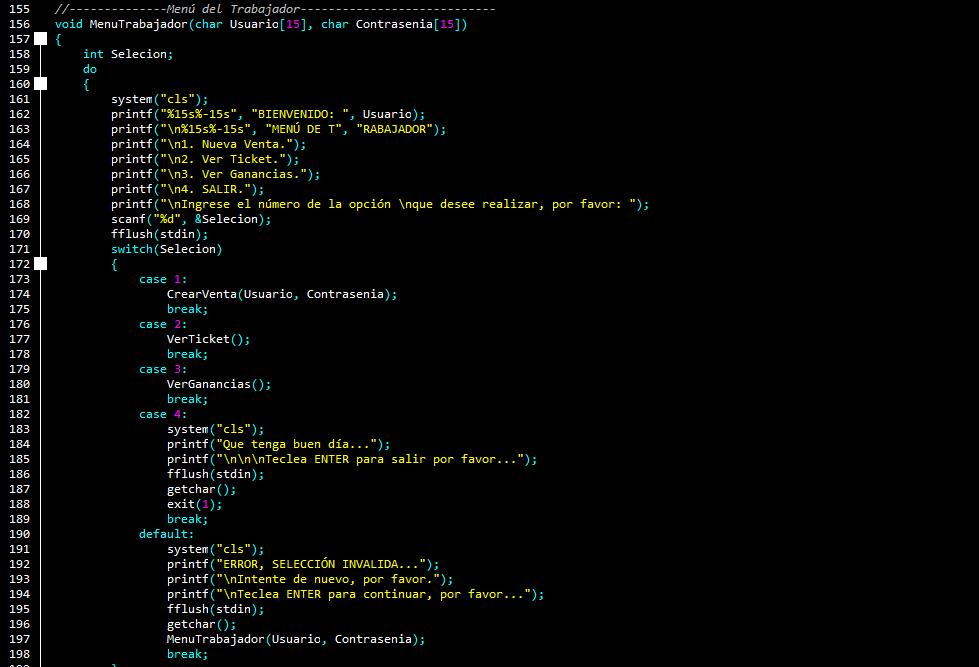
**2.- Organización del código con base en entradas, procesamiento y salidas**

**Entradas:**

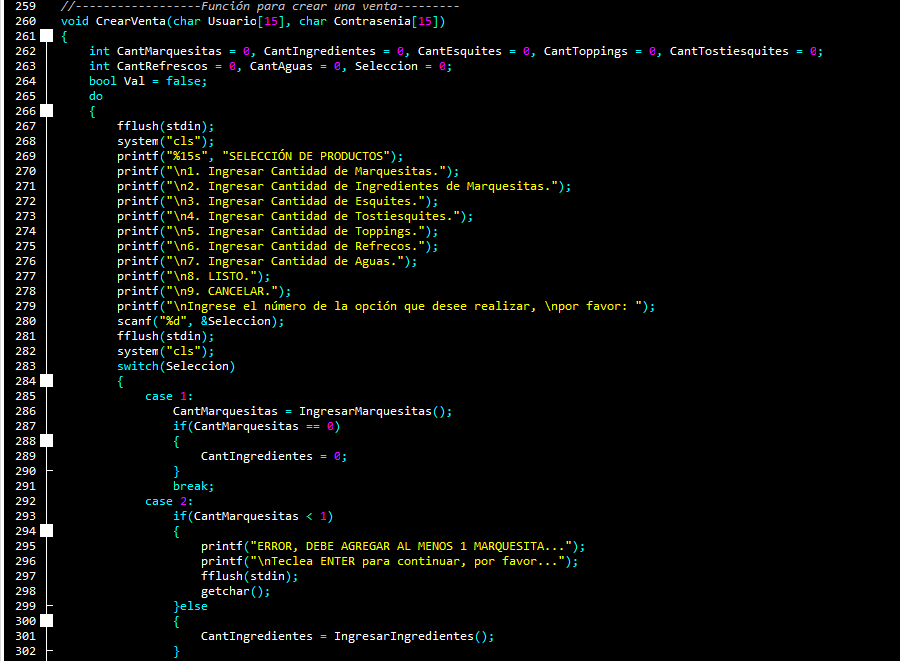
**1.- Login:**

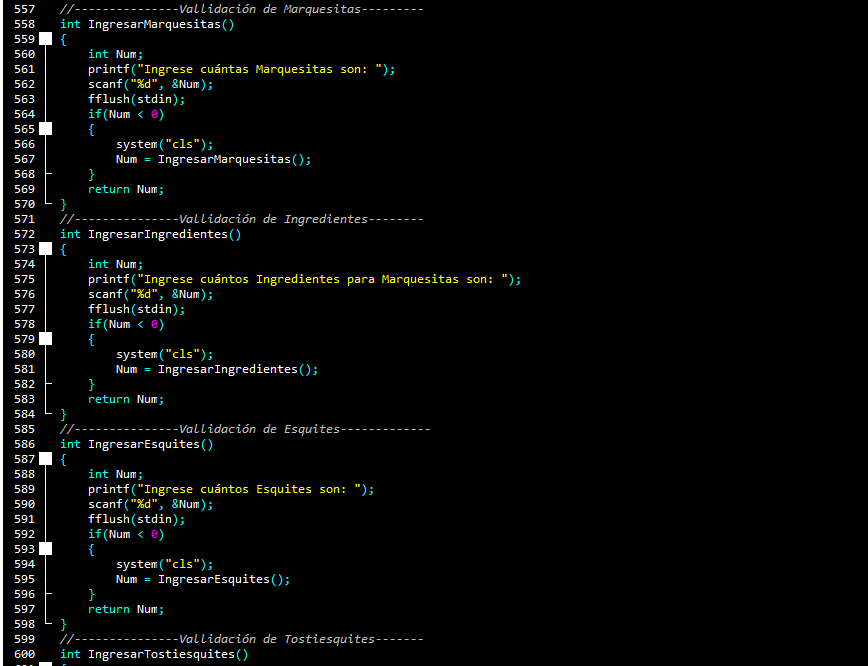
**2.- Menús**

Administrador:

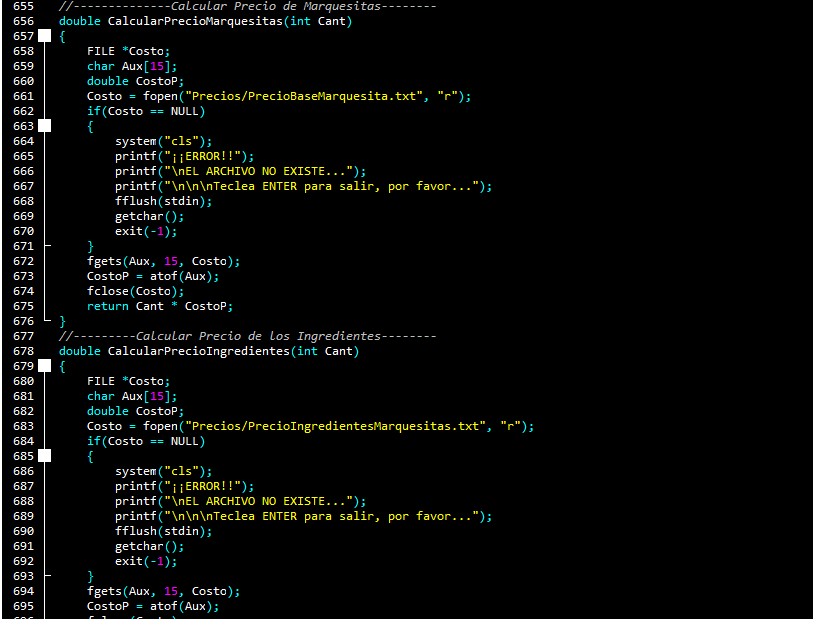
Empleado:

**Procesos:**

**1.- Ventas:**

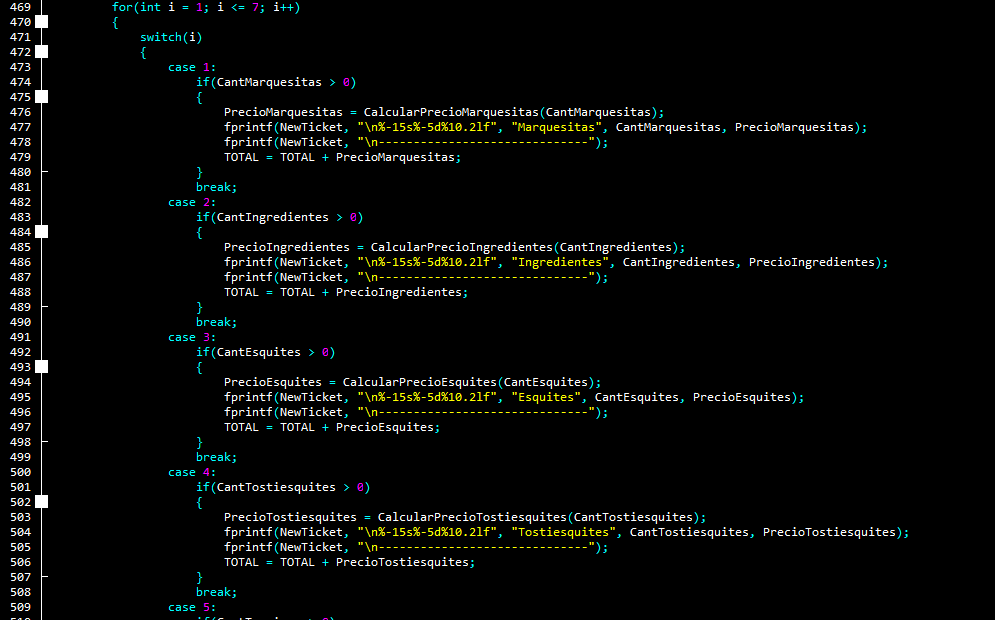
**2.- Validaciones de los productos:**

**Salidas:**

**1.- Creaciones de los tickets:**

**3.- Cohesión de las funciones**

**Invocación de funciones:**

**Pasó de parámetros:**

**Proceso de Desarrollo**

**1.- Trabajo de codificación acorde con la propuesta de herramientas, forma de organización, esquemas de monitoreo y bitácoras.**

* Como herramienta de código se esta utilizando el programa Dev-C++.
* Nos estamos organizando a través de plataformas virtuales y uso de mensajería instantánea para la aclaración de dudas.
* Se esta implementando una bitácora para llevar el control del proyecto que sería la siguiente:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Actividades** | **Sesión 1** | **Sesión 2** | **Sesión 3** | **Sesión 4** | **Sesión 5** | **Sesión 6** | **Sesión 7** | **Sesión 8** | **Sesión 9** | **Sesión 10** |
| Elección del proyecto |  |  |  |  |  |  |  |  |  |  |
| Planificación de la estructura del proyecto. |  |  |  |  |  |  |  |  |  |  |
| Investigación de antecedentes de la propuesta. |  |  |  |  |  |  |  |  |  |  |
| Partición del proyecto (modularización) |  |  |  |  |  |  |  |  |  |  |
| Codificación |  |  |  |  |  |  |  |  |  |  |
| Revisión de los avances previo a la entrega |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |
| Entrega final |  |  |  |  |  |  |  |  |  |  |

***Observaciones:***

* Las sesiones previas a la **sesión 3**, son fechas previas a la entrega de la propuesta el lunes 13 de abril del 2020.
* Las **sesiones de la 3 a la 7** son fechas entre el 14 y 30 de abril del 2020, siendo el 30 de abril la fecha de entrega de los avances del proyecto.
* Las sesiones de la 8 a la 10 serían fechas entre el 1 y 11 de mayo del 2020, siendo el 11 de mayo la fecha programada de la entrega final.

**2.- Evaluación basada en objetivos (medibles) de la participación individual y de grupo.**

|  |  |  |
| --- | --- | --- |
| **Nombre** | **Contribuciones** | **Participación** |
| Alan Josue Barón Pat | Casos de uso, proceso de desarrollo, bitácoras. | 50% |
| Victor Enrique Cauich Davalos | Idea principal, requerimientos del sistema, modularidad. | 50% |
|  |  | **Total**: 100% |