07/04/2023

**EA**S**Y SHOP**

# Índice

[**Índice 1**](#_4sygxixpgan2)

[**2.- Resumen ejecutivo 3**](#_72k5poh99yiu)

[**3.- Datos descriptivos 3**](#_rwyrxudxvo6t)

[3.1.- Descripción general 3](#_9x2sh48hbcwt)

[3.2.- Marco teórico 4](#_cxaa2iup2ou6)

[3.3.- Principios tecnológicos y normas técnicas aplicables 4](#_56108k1ndqgl)

[3.4.- Análisis de la realidad 4](#_unp5l8boa3cg)

[3.5.- Entorno de la empresa colaboradora 5](#_fmntznlayfg9)

[3.6.- Justificación (Necesidad/Innovación detectada) 5](#_ql2uygyc9c3e)

[3.7.- Destinatarios 6](#_ogurwkkut4c1)

[3.8.- Capa tecnológica 6](#_a4ogdef8p4mn)

[3.8.1.- Base de datos 6](#_5ay6g815zfha)

[3.8.1.1.- SQL Server 6](#_lmubulj4go8e)

[3.8.1.2.- SQL Profile 7](#_rciky5d4f39k)

[3.8.2.- Lenguaje de programación 7](#_aa6jftidu1d)

[3.8.1.- C# 7](#_cyr4rsxgoeup)

[**4.- Desarrollo del proyecto 7**](#_vx0ulqnboa3o)

[4.1.- Acuerdo de proyecto 7](#_rzwx1hnevqol)

[4.1.1.- Necesidad o innovación surgida 7](#_pctoe9n2w77)

[4.1.2.- Requisitos funcionales 7](#_8kbdxnotz4sh)

[4.1.3.- Requisitos no funcionales 8](#_2v5wf3i4ftf3)

[4.2.- Actividades y cronología 8](#_h7s8t0o7c3bp)

[4.2.1.- Diagrama de Gantt 8](#_rdprc1amvsdl)

[4.2.2.- Documento de análisis y diseño 9](#_vfe17z868fj0)

[4.2.2.1.- Diagrama de Bases de datos 9](#_1rw0qk8w4d85)

[4.2.2.2.- Diagrama de casos de uso 11](#_pezfb3nlfsbl)

[4.2.2.2.1.- Usuario 12](#_vnqrdwttm94q)

[4.2.2.2.2.- Usuario invitado 17](#_sonvrxthynt2)

[4.2.2.3.- Diagrama de clases 21](#_81qvhfd3txv7)

[4.2.2.4.- Diagrama de secuencia 22](#_79n82cnrttq6)

[4.2.2.4.1.- Usuario 22](#_m3b89iowsstz)

[4.2.2.4.2.- Usuario invitado 24](#_h4z21rsdxyl0)

[4.2.3.- Pruebas del sistema 25](#_4uuvvybhhv0j)

[4.2.3.1.- SQL Server Profile 25](#_xgxg867lbd0p)

[4.2.3.2.- MSTest 25](#_a07njqh9vmdt)

[4.2.4.- Documentación 28](#_g26sgdwbn2g6)

[4.2.4.1.- Manual de usuario 28](#_gk1ayal1m11i)

[4.2.4.1.1.- Catálogo de categorías y subcategorías 28](#_ocvuxikienco)

[4.2.4.1.2.- Registro 29](#_rlf797rooqxe)

[4.2.4.1.3.- Verificar cuenta 29](#_wwlyi5pqmtwq)

[4.2.4.1.4.- Inicio sesión 30](#_hwb0z6aed9zi)

[4.2.4.1.5.- Recuperar contraseña 30](#_fh0bqfnxwba7)

[4.2.4.1.6.- Selección del producto 31](#_kypf2xerme27)

[4.2.4.1.7.- Agregar producto al carrito de productos 31](#_94wgwel7jblg)

[4.2.4.1.8.- Carrito de productos 32](#_mllnihqpbz97)

[4.2.4.1.9.- Confirmación de la compra 33](#_yyq9ike5luor)

[4.2.4.1.10.- Perfil 34](#_243ziqrj8js9)

[4.2.4.1.11.- Otras opciones 34](#_8i0xqens1y40)

[4.2.4.1.12.- Pagos 35](#_7zm3xtb4x2gq)

[4.2.4.1.13.- Pedidos 35](#_oswa21d5fl0z)

[4.2.4.1.14.- Más detalles 36](#_airux6x1au9k)

[4.2.4.1.15.- Direcciones 36](#_hp0extx7wkzl)

[4.2.4.1.16.- Darse de baja 37](#_ljm6uxyieiv7)

[4.2.4.2.- Manual de instalación o configuración 37](#_wyoreluw16ha)

[4.2.4.2.1.- Usuario final 37](#_99u4f1hnved8)

[4.2.4.2.2.- Desarrollador 38](#_ko2igb7zaie0)

[4.3.- Diseño de interfaz gráfica 38](#_njz9z3892yps)

[4.3.1.- Catálogo de categorías y subcategorías 39](#_gltxdcxruef8)

[4.3.2.- Registro 39](#_uvtefernn434)

[4.3.3.- Verificar cuenta 40](#_ge8igvn9fct0)

[4.3.4.- Inicio sesión 40](#_61mk8agr43ua)

[4.3.5.- Recuperar contraseña 40](#_ab6qrt5lbfds)

[4.3.6.- Selección del producto 41](#_rtq3sqxyiowi)

[4.3.7.- Agregar producto al carrito de productos 41](#_ankdpcwatz9y)

[4.3.8.- Carrito de productos 42](#_5jwexo1fc9jk)

[4.3.9.- Confirmación de la compra 43](#_x5ldr067kqq6)

[4.3.10.- Perfil 43](#_oc4bctesn88d)

[4.3.11.- Otras opciones 44](#_y8lpgjjxs2af)

[4.3.12.- Pagos 44](#_oidvixohuin)

[4.3.13.- Pedidos 45](#_yq797igrph50)

[4.3.14.- Más detalles 45](#_lbucjvnj2spa)

[4.3.15.- Direcciones 46](#_hnh2sfa7hz4r)

[4.3.16.- Darse de baja 46](#_fp2jpzh4njrb)

[4.4.- Metodología seguida 46](#_84ppnfwgm5u7)

[4.5.- Recursos 47](#_cryu6sx5k9bn)

[4.5.1.- Materiales 47](#_wt41rzu39qv9)

[4.5.2.- Infraestructuras y equipamiento 47](#_jrfb8zbn96tz)

[4.5.2.- Humanos 47](#_pyc41bxylfww)

[**5.- Presupuesto 47**](#_h2bu42u94siu)

[**6.- Resultados objetivos y conclusiones 48**](#_guciauo4laoj)

[**7.- Bibliografía 48**](#_ng49ojat3t3o)

# 2.- Resumen ejecutivo

Se presenta una innovadora aplicación de escritorio que está diseñada para brindar comodidad, ahorro de tiempo y una experiencia de compra sin complicaciones a los usuarios.

La aplicación ofrece una amplia gama de características y funcionalidades que mejoran la forma en que las personas realizan sus compras de forma online. Algunas características clave de la aplicación son las siguientes:

* Búsqueda y exploración de productos: Permite buscar y explorar fácilmente una amplia variedad de productos disponibles.
* Carrito de productos: Los productos que se desea comprar, serán recogidos en el carrito. Se le permite editar la cantidad a comprar e incluso borrarla de su carrito de productos.
* Compra en línea y entrega a domicilio: Se permite realizar la compra de forma online con entrega a domicilio.
* Integración de pago seguro: Además de posibilitar la opción de pago en efectivo, se permite realizar el pago con tarjeta de crédito. Esto brinda flexibilidad y seguridad a los clientes al realizar transacciones en línea.

An innovative desktop app is introduced that is designed to provide convenience, time savings and a seamless shopping experience to users.

The app offers a wide range of features and functionalities that improve the way people make their purchases online. Some key features of the application are as follows:

* Product Search and Exploration: Enables you to easily search and explore a wide range of available products.
* Product cart: The products you want to buy will be collected in the cart. You are allowed to edit the amount to buy and even delete it from your product cart.
* Online purchase and home delivery: The purchase is allowed online with home delivery.
* Safe payment integration: In addition to enabling the cash payment option, payment with credit card is allowed. This gives customers flexibility and security when making online transactions.

# 3.- Datos descriptivos

## 3.1.- Descripción general

Easy Shop es una aplicación de compra tipo supermercado que ofrece una experiencia de compra digitalizada. Esta aplicación busca transformar la forma en la que se compran los productos básicos, haciéndolo más conveniente y eficiente en la era digital.

## 3.2.- Marco teórico

Debido a la era digital y al crecimiento del comercio electrónico, se ha llevado a realizar esta aplicación de compra online aportando ciertas ventajas:

* Mayor alcance y accesibilidad: Permitiendo llegar a un público más amplio y diverso, debido a que el cliente puede acceder a la aplicación en cualquier lugar y momento.
* Comodidad: Permitiendo realizar la compra online desde la comodidad del hogar, oficina o cualquier otro, sin desplazarse físicamente a una tienda física.
* Mayor disponibilidad de productos: Ofrece una amplia gama de productos y variedades que no pueden estar en la tienda física.
* Personalización y recomendación: Ofrece productos y ofertas específicas basadas en el historial de compras y preferencia.
* Mayor información y comparación de productos: Proporciona mayor información detallada de los productos, reseñas de clientes y comparaciones de precios, así aumentando la confianza del usuario y fomentar la toma de decisiones informada.
* Análisis de datos: Recopila datos del usuario que pueden ser utilizados para realizar análisis y tomar decisiones informadas en cuanto a estrategias de marketing, mejoras en la experiencia del usuario.

## 3.3.- Principios tecnológicos y normas técnicas aplicables

La aplicación será desarrollada cumpliendo los principios tecnológicos y normas técnicas aplicables, protegiendo la confidencialidad, integridad y disponibilidad de los datos y sistemas. Además de ser compatible y capaz de interactuar con otros sistemas y aplicaciones de manera eficiente y efectiva, estará diseñada y desarrollada de manera que sea accesible para todas las personas. Por último, el desarrollo de esta misma se ha tenido en cuenta su impacto en el medio ambiente y en la sociedad en general.

## 3.4.- Análisis de la realidad

A pesar de que la compra online ha ganado popularidad, ofreciendo a los consumidores la conveniencia de realizar compras en línea desde la comodidad de su hogar, su implementación y adopción también han enfrentado desafíos en términos de disponibilidad de inventario, calidad de los productos, la privacidad y seguridad de los datos. En este análisis se examinará la realidad de la aplicación:

* La compra online ha ganado popularidad en los últimos años.
* Permite comparar a los usuarios los precios, leer reseñas y comparar desde casa o en cualquier momento.
* Suele ofrecer opciones de entrega a domicilio o recogida en tienda.
* Enfrenta desafíos como la disponibilidad del inventario y la calidad de los productos entregados.
* Preocupación por la privacidad y seguridad de los datos.
* Se ha generado competencia en línea y cambios en la industria de supermercados.
* Ha cambiado los hábitos de compra de los consumidores.
* Ciertos consumidores, debido a la poca privacidad o inseguridad, prefieren la experiencia de compra en tienda física.

## 3.5.- Entorno de la empresa colaboradora

Un colaborador puede ser determinante para el éxito y la efectividad de la aplicación. Hay unas consideraciones clave a tener en cuenta:

* La competencia en el mercado de venta de productos en línea puede ser alta, con varias aplicaciones y plataformas compitiendo por la atención y preferencia de los consumidores. Es importante entender y evaluar el nivel de competencia en el mercado y cómo se posiciona la empresa en este entorno.
* Las alianzas estratégicas con supermercados, distribuidores u otros actores clave en la cadena de suministro pueden ser cruciales para asegurar la disponibilidad y calidad de los productos ofrecidos a través de la aplicación. Establecer relaciones sólidas con colaboradores estratégicos puede ser un factor determinante para el éxito de la aplicación.
* Las regulaciones y el marco legal relacionados con la venta de productos en línea pueden variar según la ubicación geográfica. Es esencial comprender y cumplir con las regulaciones y requisitos legales aplicables, como los relacionados con la seguridad alimentaria, la protección del consumidor y la privacidad de datos.
* La infraestructura tecnológica es un aspecto crítico para el funcionamiento eficiente y seguro de la aplicación. Esto incluye servidores, sistemas de pago en línea, seguridad cibernética y otros elementos tecnológicos necesarios para asegurar el adecuado funcionamiento de la aplicación y proteger la información de los usuarios.
* La experiencia del usuario es fundamental para el éxito de cualquier aplicación. La empresa debe asegurarse de que la aplicación sea intuitiva, fácil de usar y ofrece una experiencia de compra en línea atractiva y conveniente para los usuarios.
* La logística y las operaciones son aspectos críticos para garantizar la eficiencia en la entrega de los productos. La empresa debe establecer un sistema eficaz de gestión de inventario, preparación de pedidos y entrega a domicilio o recogida en tienda, asegurando que los productos lleguen a los clientes de manera oportuna y en buen estado.
* Un buen servicio al cliente es esencial para mantener la satisfacción del usuario. La empresa establece canales de atención al cliente efectivos y capacitar a su personal para brindar un servicio amable y resolutivo en caso de problemas o consultas por parte de los usuarios.
* El entorno de la empresa también incluye las estrategias de marketing para promocionar la aplicación y atraer a los clientes potenciales. Esto puede incluir estrategias de publicidad en línea, promociones especiales, programas de lealtad u otras iniciativas de marketing para promover la aplicación y atraer a más usuarios.
* El mercado de venta de productos de supermercado en línea es dinámico y está en constante evolución. La empresa debe estar al tanto de las tendencias y cambios en el mercado, como la entrada de nuevos competidores, la evolución de las preferencias del consumidor y las innovaciones tecnológicas, y adaptar su estrategia en consecuencia.

## 3.6.- Justificación (Necesidad/Innovación detectada)

La justificación para una aplicación de venta de productos de supermercado se basa en la necesidad creciente de los consumidores de realizar compras en línea, incluyendo productos de supermercado, debido a los cambios en los hábitos de compra y la competencia en el mercado en línea. Además, ofrece oportunidades de establecer alianzas estratégicas con supermercados y distribuidores, brinda innovación tecnológica en el mercado minorista y responde a los cambios en la demanda del consumidor que busca comodidad y eficiencia en sus compras. Por lo tanto una aplicación debe estar bien diseñada y gestionada para así poder satisfacer estas necesidades y ofrecer una solución innovadora en el mercado de venta de productos de supermercado en línea.

## 3.7.- Destinatarios

Los destinatarios principales de la aplicación de venta de productos de supermercado son los usuarios consumidores que buscan una forma conveniente, eficiente y segura de realizar sus compras de productos de supermercado en línea. Estos usuarios pueden ser personas ocupadas que buscan ahorrar tiempo y esfuerzo en sus compras, personas que prefieren la comodidad de hacer compras desde su hogar o lugar de trabajo, o aquellos que buscan opciones de compra personalizadas y flexibles.

Además, la aplicación también puede dirigirse a usuarios consumidores que buscan productos específicos, como alimentos orgánicos, productos locales, opciones de dieta especializada, entre otros. También puede atraer a usuarios consumidores que buscan ofertas y descuentos exclusivos, así como a aquellos que buscan una experiencia de compra en línea fácil de usar y segura.

En resumen, los destinatarios de la aplicación de venta de productos de supermercado son usuarios consumidores que buscan una solución conveniente, eficiente y personalizada para realizar sus compras de supermercado en línea, independientemente de sus preferencias de compra y necesidades específicas.

## 3.8.- Capa tecnológica

### 3.8.1.- Base de datos

#### 3.8.1.1.- SQL Server

Para el desarrollo de la aplicación, se hará uso de la base de datos Sql Server debido a que se ha visto tanto en clase como en el trabajo.

Para las consultas, se hará uso de procedimientos almacenados, así se separa código de sql y código de programación. La base de datos estará formada por todas las tablas necesarias para el desarrollo de la aplicación.

Más adelante se adjunta un [diagrama de la base de datos](#_1rw0qk8w4d85) que plasmará todas las tablas usadas para la aplicación.

#### 3.8.1.2.- SQL Profile

Para controlar el envío de datos desde la aplicación a la base de datos, se hace uso de esta herramienta. Para más información de su uso, mirar el apartado [Pruebas del sistema](#_xgxg867lbd0p).

### 3.8.2.- Lenguaje de programación

#### 3.8.1.- C#

Como lenguaje de programación se ha elegido C# por las mismas razones que se ha elegido Sql Server. Mediante código C# se realizarán todas las ventanas, controles, propiedades para el desarrollo de la aplicación.

# 4.- Desarrollo del proyecto

## 4.1.- Acuerdo de proyecto

### 4.1.1.- Necesidad o innovación surgida

Debido al crecimiento de la demanda por parte de los consumidores por hacer sus compras de manera más eficiente y conveniente,surgió la necesidad de crear esta aplicación. En la actualidad, la mayoría de las personas tienen estilos de vida ocupados y no tienen el tiempo para ir al supermercado a realizar sus compras. Además, desde la pandemia de COVID-19 se ha llevado a buscar alternativas más seguras para hacer la compra.

La innovación que impulsa la necesidad de crear esta aplicación, es el uso de la tecnología y la conectividad a Internet permitiendo a los consumidores realizar compras en línea y recibir sus productos en la comodidad de sus hogares. La aplicación ofrece una plataforma en línea fácil de usar, para una compra rápida y eficiente, seleccionando los productos que necesita y realizando el pago en línea.

Además, las aplicaciones en línea pueden ofrecer beneficios adicionales, como la posibilidad de comparar precios, recibir ofertas y descuentos exclusivos, recibir recomendaciones personalizadas en función de las compras realizadas.

### 4.1.2.- Requisitos funcionales

Los requisitos funcionales de la aplicación podrían ser los siguientes:

* Registro de usuario: El consumidor se registra en la aplicación, realiza su compra y almacena sus datos.
* Modo invitado: La compra se realiza sin la necesidad de registrarse. Sus datos no son almacenados permanentemente o hasta que se dé de baja de la aplicación, se borran en un determinado tiempo.
* Catálogo de productos: Al iniciar la aplicación se muestra un listado completo de las categorías disponibles.
* Búsqueda y filtrado: Se permitirá al usuario buscar y filtrar productos.
* Carrito de compra: El usuario podrá agregar sus productos en un carrito y editar la cantidad de productos en el carrito.
* Pago en línea: El pago se realizará en línea mediante un pago seguro.
* Historial de compras: Se proporciona un historial de compras realizadas anteriormente.

### 4.1.3.- Requisitos no funcionales

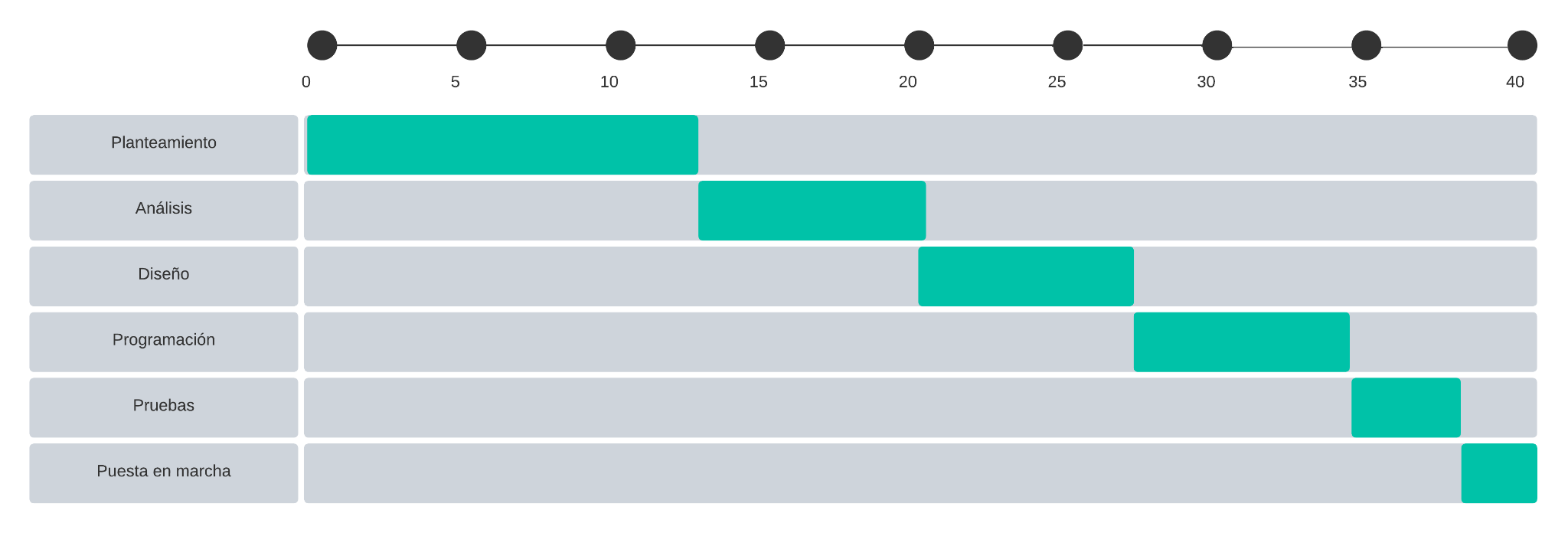
Los requisitos no funcionales de la aplicación podrían ser los siguientes:

* Seguridad: Se garantiza la seguridad de los datos de los usuarios y la información de pago.
* Rendimiento: Se asegura que la aplicación es rápida y eficiente.
* Usabilidad: Se asegura que la aplicación es fácil de usar y accesible para los usuarios.
* Escalabilidad: Se permite que la aplicación pueda manejar un alto volumen de tráfico de usuarios sin afectar su rendimineto.
* Disponibilidad: La aplicación estará disponible en todo momento.
* Compatibilidad: La aplicación será compatible para los diferentes dispositivos.

## 4.2.- Actividades y cronología

### 4.2.1.- Diagrama de Gantt

Para la planificación y el desarrollo del proyecto, se hará uso del diagrama de Gantt. Este diagrama muestra el calendario de las tareas del proyecto en el eje vertical y el tiempo en el eje horizontal. Además, nos permitirá visualizar las dependencias entre tareas, tiempo estimado de finalización y la secuencia de actividades.



Como se puede apreciar, el proyecto se divide en seis etapas:

* Planteamiento: Se define la función que debe llevar a cabo el software. Se identifican las necesidades y requisitos del usuario final. Por último, se establecen los objetivos y alcances del proyecto.
* Análisis: Se hace un análisis más profundo de los requisitos del usuario final. Se establecen los requisitos funcionales (la función del software) y no funcionales (características del software).
* Diseño: Se diseña la arquitectura del software. Se elaboran los diagramas necesarios para el desarrollo de esta misma (más adelante), los cuales describen el comportamiento y la funcionalidad del software. Por último, se decide el lenguaje de programación para el desarrollo.
* Programación: Se lleva a cabo la implementación del software según el plan de diseño. Se escribe el código de programación para crear el software.
* Pruebas: Se realizan las pruebas necesarias para garantizar que el software cumple con lo planeado y funciona correctamente.
* Puesta en marcha: Se instala el software en los sistemas. Se realiza una verificación final para asegurarse de que el software está funcionando correctamente y que cumple con los requisitos del usuario final.

Otros puntos a destacar:

* Diseño:

Debido a tener que buscar información para ciertos aspectos de la aplicación, el tiempo estimado para la parte del diseño se tuvo que alargar.

Además, la base de datos es bastante extensa, por lo que la búsqueda de productos e imágenes llevó tiempo extra.

Aproximadamente 30 horas más de las establecidas.

* Programación:

Al tener dos posibilidades de realizar la compra, como usuario o como invitado, toca ir cambiando constantemente ciertas partes del código para no saltar a error, incluso a investigar una manera para realizar la compra de forma de invitado para no hacerla desde ficheros de texto.

Por último, la búsqueda de información respecto al envío de correos y el mensaje con un formato tipo HTML y los reportes de compra, han llevado tiempo extra al estimado de la parte de programación.

Aproximadamente, más de 40 horas de las establecidas.

* Pruebas:

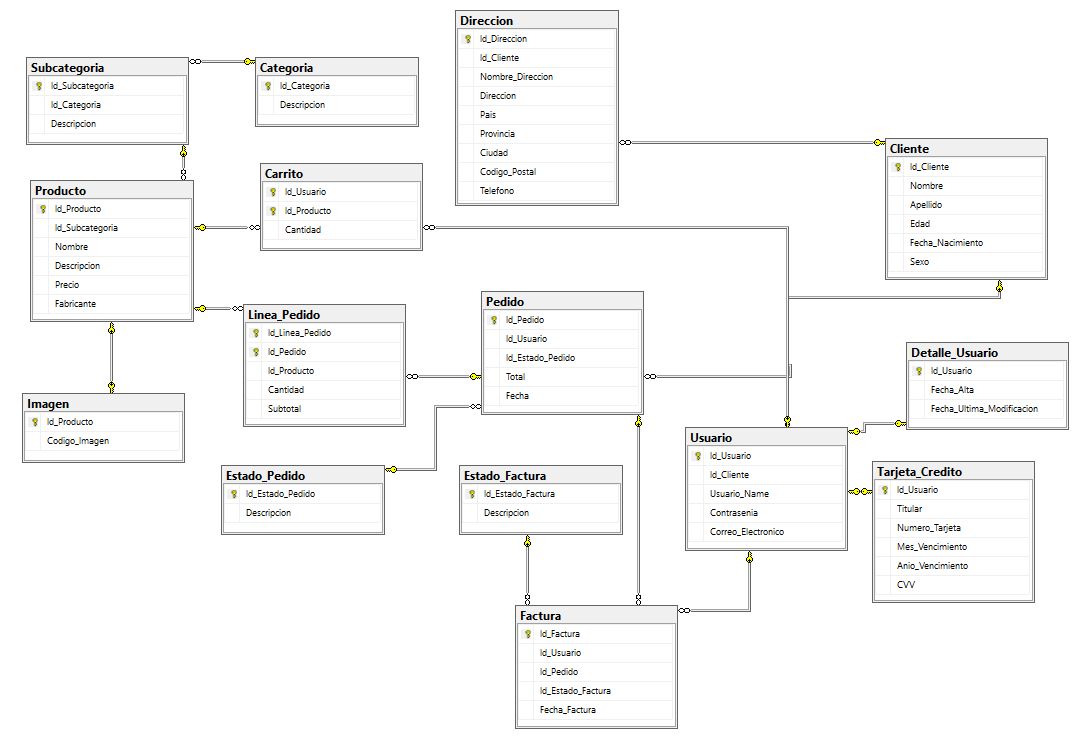
Como ocurre con los puntos anteriores, debido a la búsqueda de información respecto a las pruebas, el tiempo estimado de esta misma, fue prolongado.

Aproximadamente, 5 horas más de las establecidas.

### 4.2.2.- Documento de análisis y diseño

#### 4.2.2.1.- Diagrama de Bases de datos

Para el desarrollo de la aplicación se hará uso de las siguientes tablas:



Tablas principales:

* Cliente: guarda información sobre los clientes.

Incluye: Id\_Cliente, Nombre, Apellido, Edad, Fecha\_Nacimiento, Sexo.

* Direccion: guarda información sobre la dirección del cliente.

Incluye: Id\_Direccion, Id\_Cliente, Nombre, Direccion, Pais, Provincia, Ciudad, Codigo\_Postal, Telefono.

* Usuario: guarda información del usuario.

Incluye: Id\_Usuario, Id\_Cliente, Usuario\_Name, Contrasenia, Correo\_Electronico, Invitado.

* Tarjeta\_Credito: Representa la tarjeta de crédito asociada al usuario.

Incluye: Id\_Usuario, Titular, Numero\_Tarjeta, Mes\_Vencimiento, Anio\_Vencimiento, CVV.

* Producto: guarda información sobre los productos.

Incluye: Id\_Producto, Id\_Subcategoria, Nombre, Descripcion, Precio, Fabricante.

* Estado\_Pedido: guarda información sobre el estado del pedido.

Incluye: Id\_Estado\_Pedido, Descripcion.

* Pedido: guarda información sobre los pedidos realizados por los clientes.

Incluye: Id\_Pedido, Id\_Usuario, Id\_Direccion, Id\_Estado\_Pedido, Fecha, Id\_Metodo\_Pago.

* Linea\_Pedido: guarda información sobre los productos incluidos en un pedido.

Incluye: Id\_Linea\_Pedido, Id\_Pedido, Id\_Producto, Cantidad, Subtotal.

* Categoria: La categoría a la que pertenece cada producto. Hay nueve categorías.

Incluye: Id\_Categoria. Descripcion.

* Subcategoria: La categoría se compone de subcategorías. En total hay 53 subcategorías. Hay seis subcategorías para cada categoría.

Incluye: Id\_Subcategoria, Id\_Categoria, Descripcion.

* Carrito: Almacena temporalmente los productos añadidos por los clientes hasta que se confirme la compra.

Incluye: Id\_Usuario, Id\_Producto, Cantidad.

* Estado\_Pedido: Informa el estado del pedido: En proceso, de camino y entregado.

Incluye: Id\_Estado\_Pedido, Descripcion.

* Factura: Sirve como resumen detallada de la compra, establece las condiciones de pago.

Incluye: Id\_Factura, Id\_Pedido, Id\_Estado\_Factura, Fecha\_Factura.

* Estado\_Factura: Informa el estado de la factura: Pendiente y cobrado.

Incluye: Id\_Estado\_Factura, Descripcion.

Otras tablas:

* Detalle\_Usuario: Informa la fecha de alta del usuario y la última fecha que se actualizó los datos.

Incluye: Id\_Usuario, Fecha\_Alta, Fecha\_Ultima\_Modificacion.

* Imagen: Guarda las referencias de imágenes para cada producto. Esta referencia será recogida para mostrar la imagen correspondiente a su producto al iniciar la aplicación.

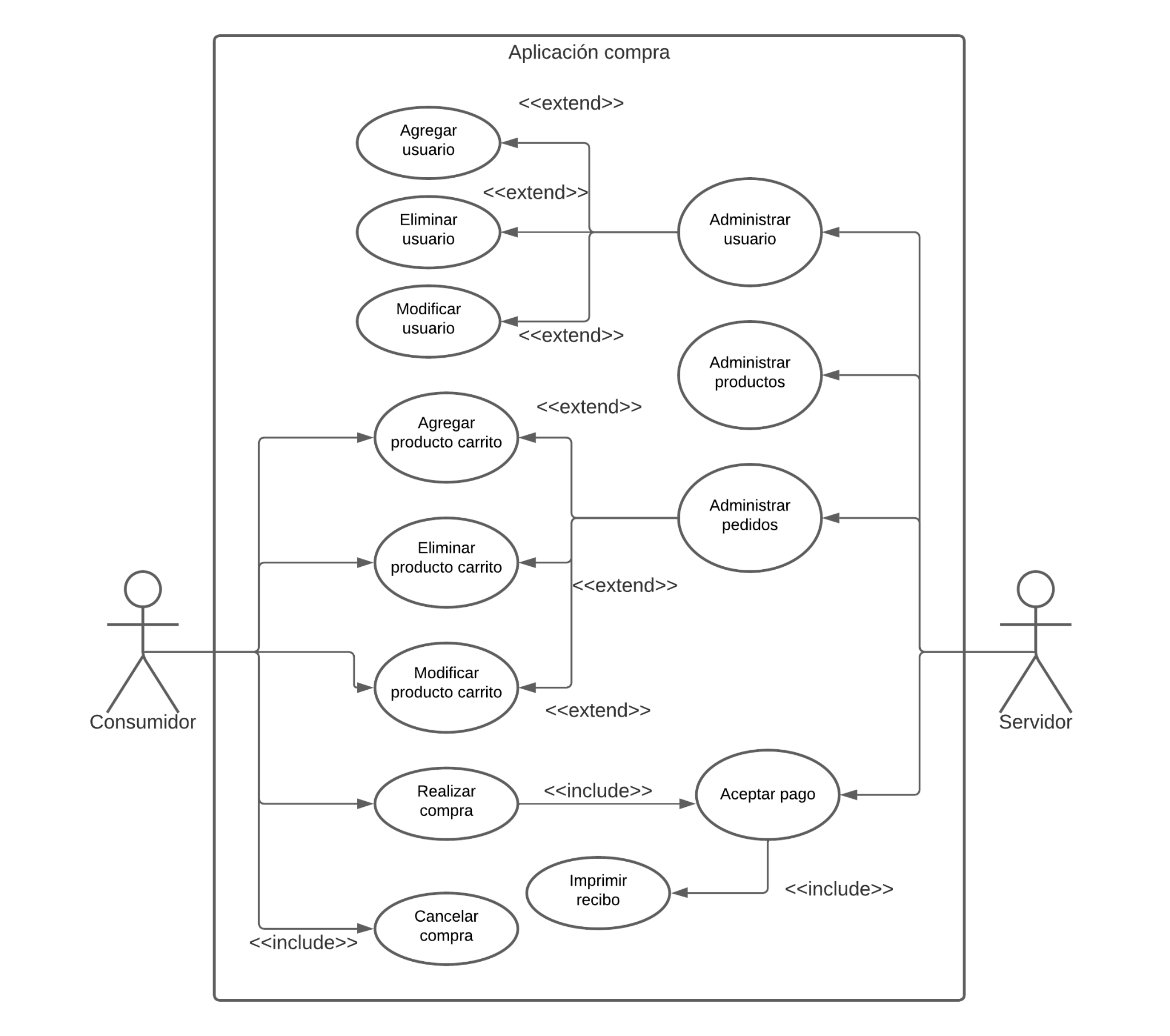
Incluye: Id\_Producto, Codigo\_Imagen.

#### 4.2.2.2.- Diagrama de casos de uso

Mediante este diagrama, la intención es identificar y definir las funcionalidades que el sistema debe proporcionar para satisfacer las necesidades del usuario. Además, nos permitirá comprender cómo el usuario interactúa con el sistema y qué funciones son importantes para ellos.

A Continuación se muestra el diagrama con sus respectivos comentarios:

##### 4.2.2.2.1.- Usuario



| Nº | Campo | Agregar usuario |
| --- | --- | --- |
| 1 | Actor | Servidor |
| 2 | Tipo de caso de uso | Base |
| 3 | Breve descripción | El cliente se da de alta y el sistema lo acepta. |
| 4 | Precondiciones | El cliente rellena el formulario de registro y está de acuerdo con la política de la aplicación y cómo serán tratados sus datos. |
| 5 | Flujo de eventos | - El consumidor accede a la aplicación.  - Accede al área de registro.  - Rellena los datos obligatorios.  - Acepta el registro.  - El sistema comprueba lo necesario para darlo de alta o no.  - Si no hay ningún conflicto, será dado de alta en el sistema sin problema. |
| 6 | Postcondiciones | Una vez rellenado y comprobado los datos del usuario y su posible existencia en base de datos, sería dado de alta en el sistema. |
| 7 | Requisitos | No estar dado de alta ya con el nombre de usuario dado en el formulario de registro. |
| 8 | Observaciones |  |

| Nº | Campo | Eliminar usuario |
| --- | --- | --- |
| 1 | Actor | Servidor |
| 2 | Tipo de caso de uso | Base |
| 3 | Breve descripción | El cliente quiere darse de baja del sistema. |
| 4 | Precondiciones | Estar dado de alta en el sistema. |
| 5 | Flujo de eventos | - El usuario accede en el área de darse de baja.  - El sistema comprueba que el usuario dice ser quien es para dar la baja.  - Una vez comprobado con la contraseña de acceso, será borrado del sistema. |
| 6 | Postcondiciones | Se da de baja del sistema al usuario. |
| 7 | Requisitos | Estar dado de alta para poder ser dado de baja. |
| 8 | Observaciones |  |

| Nº | Campo | Modificar usuario. |
| --- | --- | --- |
| 1 | Actor | Servidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El usuario desea editar algún dato suyo. |
| 4 | Precondiciones | Estar dado de alta en el sistema. |
| 5 | Flujo de eventos | - El usuario entra en el área de editar los datos.  - Rellena el dato que el sistema le permite modificar.  - El sistema acepta el cambio. |
| 6 | Postcondiciones | El sistema edita los nuevos datos y los muestra. |
| 7 | Requisitos | Estar dado de alta en el sistema. |
| 8 | Observaciones |  |

| Nº | Campo | Agregar producto carrito. |
| --- | --- | --- |
| 1 | Actor | Consumidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El usuario agrega al carrito el producto que desea comprar. |
| 4 | Precondiciones | El producto tiene que estar disponible en stock. |
| 5 | Flujo de eventos | - El usuario agrega al carrito el producto.  - El sistema almacena el producto en una tabla ‘Carrito’. Almacena lo necesario para identificar el carrito de cada usuario.  - El contador del carrito aumenta.  - El total y subtotal se calcula. |
| 6 | Postcondiciones | Producto añadido al carrito. |
| 7 | Requisitos | Producto disponible para ser comprado. |
| 8 | Observaciones |  |

| Nº | Campo | Eliminar producto carrito. |
| --- | --- | --- |
| 1 | Actor | Consumidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El usuario desea eliminar un producto de su carrito de productos a comprar. |
| 4 | Precondiciones | El producto tiene que estar agregado en el carrito de productos para ser borrado de ahí. |
| 5 | Flujo de eventos | - El usuario elimina el producto que está en su carrito de productos.  - Por lo tanto, el producto de la tabla ‘Carrito’ también es eliminado del sistema.  - El contador del carrito disminuye. |
| 6 | Postcondiciones | El producto es eliminado del carrito de productos. |
| 7 | Requisitos | El producto debe estar agregado al carrito de productos. |
| 8 | Observaciones |  |

| Nº | Campo | Modificar producto carrito. |
| --- | --- | --- |
| 1 | Actor | Consumidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El usuario desea modificar la cantidad de un producto añadido al carrito de productos. |
| 4 | Precondiciones | El producto debe estar agregado en el carrito de productos. |
| 5 | Flujo de eventos | - El usuario modifica la cantidad de un producto del carrito de productos. También se modifica de la tabla ‘Carrito’.  - El total y subtotal se recalculan. |
| 6 | Postcondiciones | La cantidad añadida de un producto cambia. |
| 7 | Requisitos | Estar agregado en el carrito de productos. |
| 8 | Observaciones |  |

| Nº | Campo | Realizar compra. |
| --- | --- | --- |
| 1 | Actor | Consumidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El usuario desea realizar la compra. |
| 4 | Precondiciones | En el carrito de productos debe haber como mínimo un producto añadido. |
| 5 | Flujo de eventos | - El usuario no selecciona más productos.  - Accede al área de pago.  - Está de acuerdo con el precio a pagar y confirma la compra.  - El sistema comprueba el total a cobrar.  - Cobra el importe total.  - Se almacena un pedido con su detalle pedido correspondiente.  - La tabla ‘Carrito’ se vacía.  - Se almacena una factura con referencia al usuario, al pedido.  - Se da la posibilidad de imprimir o enviar la factura al correo electrónico. |
| 6 | Postcondiciones | Compra realizada con éxito. |
| 7 | Requisitos | Tener como mínimo un producto añadido. |
| 8 | Observaciones |  |

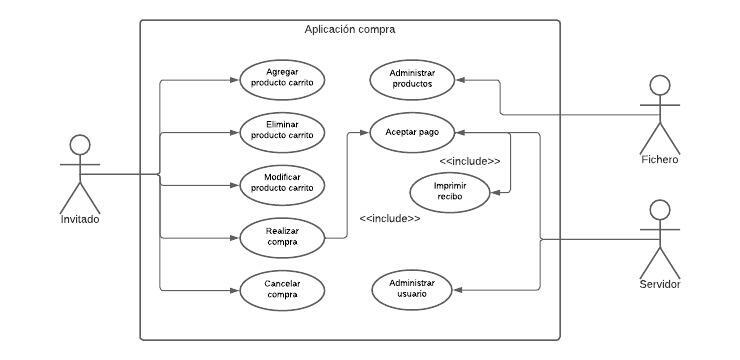
| Nº | Campo | Cancelar compra. |
| --- | --- | --- |
| 1 | Actor | Consumidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El usuario no desea realizar la compra. |
| 4 | Precondiciones | Tener producto a comprar y estar en el área de pago. |
| 5 | Flujo de eventos | - En el momento de confirmar la compra, el usuario está en desacuerdo con el precio total a pagar o simplemente decidió borrar algún producto o incluso seguir comprando.  - Finalmente decide cancelar la compra. |
| 6 | Postcondiciones | Se cancela la confirmación de la compra. |
| 7 | Requisitos | Tener productos a comprar. |
| 8 | Observaciones |  |

| Nº | Campo | Aceptar pago. |
| --- | --- | --- |
| 1 | Actor | Servidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El sistema comprueba si aceptar o no la compra. |
| 4 | Precondiciones | Tener productos en el carrito para confirmar la compra. |
| 5 | Flujo de eventos | - En el área de confirmar la compra, el usuario está de acuerdo con el total a pagar.  - El sistema comprueba que los datos de pago son correctos.  - Recibe el método de pago: Tarjeta o efectivo.  - Si el método es la tarjeta, se realiza el cobro a la tarjeta.  - Si es en efectivo, la factura estará pendiente. |
| 6 | Postcondiciones | Compra realizada con éxito. |
| 7 | Requisitos | Tener productos a comprar en el carrito. |
| 8 | Observaciones |  |

| Nº | Campo | Imprimir recibo. |
| --- | --- | --- |
| 1 | Actor | Servidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | Una vez confirmada y realizada la compra, el sistema muestra una factura detallada de la compra realizada. |
| 4 | Precondiciones | Haber confirmado y cobrado el total de la compra. |
| 5 | Flujo de eventos | Una vez realizada la compra con éxito, el sistema muestra una factura detallada la cual puede ser guardada o impresa por el usuario. |
| 6 | Postcondiciones | Se muestra la factura detallada. |
| 7 | Requisitos | Haber cobrado el total de la compra. |
| 8 | Observaciones |  |

##### 

##### 4.2.2.2.2.- Usuario invitado



| Nº | Campo | Agregar producto carrito. |
| --- | --- | --- |
| 1 | Actor | Invitado |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El producto elegido por el usuario invitado, se agrega al fichero carrito. |
| 4 | Precondiciones | Iniciar la aplicación. |
| 5 | Flujo de eventos | Iniciada la aplicación. Se elige el producto a agregar, este se almacena temporalmente en el fichero correspondiente. |
| 6 | Postcondiciones | Producto agregado. El contador del carrito aumenta. |
| 7 | Requisitos | Iniciada la aplicación. |
| 8 | Observaciones |  |

| Nº | Campo | Eliminar producto carrito. |
| --- | --- | --- |
| 1 | Actor | Invitado |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El producto elegido por el usuario invitado, se elimina del fichero carrito. |
| 4 | Precondiciones | Iniciar la aplicación. Posicionarse en la ventana Carrito. |
| 5 | Flujo de eventos | Iniciada la aplicación, desde la ventana Carrito se elige el producto a eliminar. |
| 6 | Postcondiciones | Producto eliminado. El contador del carrito disminuye. |
| 7 | Requisitos | Iniciada la aplicación y tener productos agregados. |
| 8 | Observaciones |  |

| Nº | Campo | Modificar producto carrito. |
| --- | --- | --- |
| 1 | Actor | Invitado |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | Se modifica la cantidad elegida del producto en el carrito. |
| 4 | Precondiciones | Iniciar la aplicación. Tener productos en el carrito. |
| 5 | Flujo de eventos | Iniciada la aplicación. Posicionado en la ventana Carrito, se modifica la cantidad del producto. |
| 6 | Postcondiciones | Producto agregado. La cantidad del producto se modifica. |
| 7 | Requisitos | Iniciada la aplicación. |
| 8 | Observaciones |  |

| Nº | Campo | Realizar compra. |
| --- | --- | --- |
| 1 | Actor | Invitado |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | La compra se realiza. |
| 4 | Precondiciones | Iniciar la aplicación. Acceder al área de acceso pago. |
| 5 | Flujo de eventos | Una vez conforme con los productos a comprar, se rellenan los datos necesarios para realizar la compra. |
| 6 | Postcondiciones | Rellenar todos los datos necesarios para la compra. |
| 7 | Requisitos | Estar conforme con los productos elegidos y rellenar todos los datos necesarios para realizar la compra. |
| 8 | Observaciones |  |

| Nº | Campo | Cancelar compra. |
| --- | --- | --- |
| 1 | Actor | Invitado |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | Se cancela la compra. |
| 4 | Precondiciones | Tener productos agregados y posicionarse en el área de acceso pago. |
| 5 | Flujo de eventos | Al no estar conforme con los productos elegidos, se cancela la compra. |
| 6 | Postcondiciones | Se regresa a la ventana Carrito. |
| 7 | Requisitos | Tener productos en el carrito. |
| 8 | Observaciones |  |

| Nº | Campo | Administrar productos. |
| --- | --- | --- |
| 1 | Actor | Fichero. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | Administra los productos agregados al fichero carrito. |
| 4 | Precondiciones | Haber agregado producto al carrito. |
| 5 | Flujo de eventos | El producto elegido se almacena en el fichero carrito según cómo esté configurado el fichero. |
| 6 | Postcondiciones | El producto se almacena en el fichero carrito. |
| 7 | Requisitos | Agregar un producto. |
| 8 | Observaciones |  |

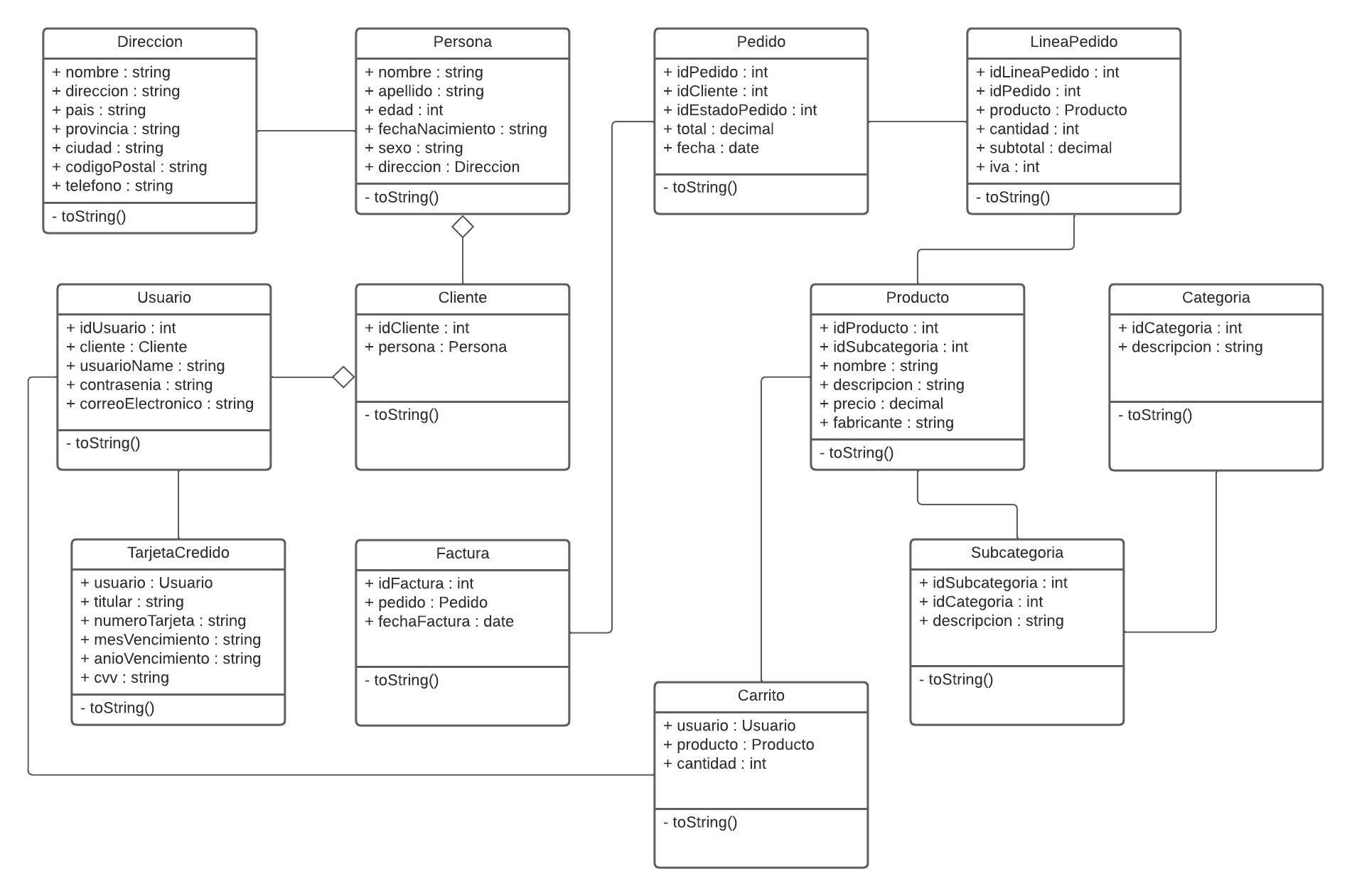
| Nº | Campo | Aceptar pago. |
| --- | --- | --- |
| 1 | Actor | Servidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | El sistema comprueba si aceptar o no la compra. |
| 4 | Precondiciones | Tener todos los datos necesarios para realizar la compra. |
| 5 | Flujo de eventos | - En el área de confirmar la compra, el usuario está de acuerdo con el total a pagar.  - El sistema comprueba que los datos de pago son correctos.  - Recibe el método de pago: Tarjeta o efectivo.  - Si el método es la tarjeta, se realiza el cobro a la tarjeta.  - Si es en efectivo, la factura estará pendiente. |
| 6 | Postcondiciones | Compra realizada con éxito. |
| 7 | Requisitos | Tener productos a comprar en el carrito. |
| 8 | Observaciones |  |

| Nº | Campo | Imprimir recibo. |
| --- | --- | --- |
| 1 | Actor | Servidor. |
| 2 | Tipo de caso de uso | Base. |
| 3 | Breve descripción | Una vez confirmada y realizada la compra, el sistema muestra una factura detallada de la compra realizada. |
| 4 | Precondiciones | Haber confirmado y cobrado el total de la compra. |
| 5 | Flujo de eventos | Una vez realizada la compra con éxito, el sistema muestra una factura detallada la cual puede ser guardada o impresa por el usuario. |
| 6 | Postcondiciones | Se muestra la factura detallada. |
| 7 | Requisitos | Haber cobrado el total de la compra. |
| 8 | Observaciones |  |

##### 

#### 4.2.2.3.- Diagrama de clases

Mediante el diagrama de clases representamos de forma visual las clases que se emplearán en el sistema y las relaciones entre ellas:



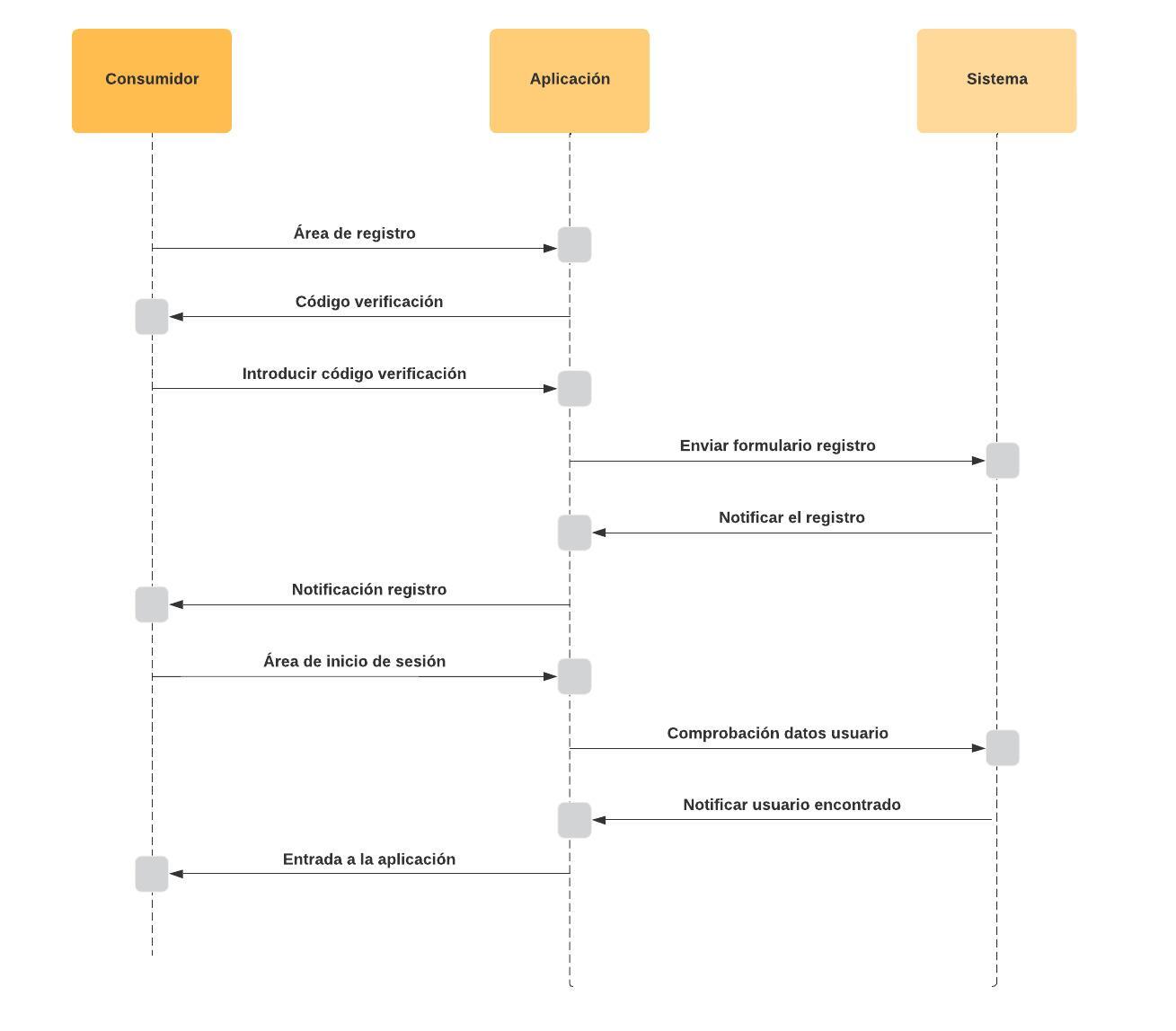
Estas clases tienen los mismos atributos que las de la base de datos. [Diagrama de base de datos.](#_1rw0qk8w4d85)

#### 4.2.2.4.- Diagrama de secuencia

Mediante este diagrama podemos visualizar la interacción entre los objetos del sistema y en qué orden. A continuación se explicarán los diagramas de secuencia según el modo de acceso:

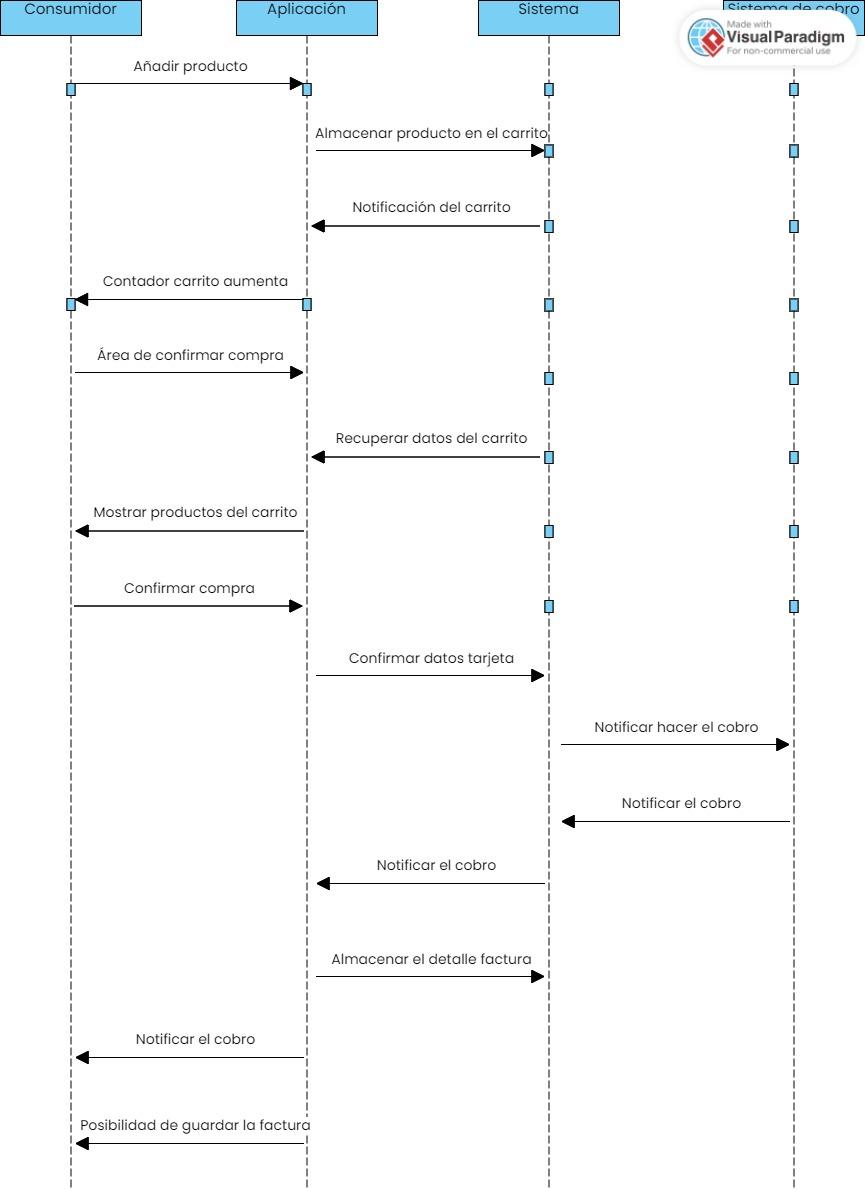
##### 4.2.2.4.1.- Usuario

* Diagrama de secuencia: Iniciar sesión y registro



1. El consumidor accede a la aplicación. Accede al área de inicio de sesión/Registro.
2. Como cliente nuevo procederá a rellenar el formulario de registro.
3. Se envía un código al correo proporcionado. Este se ingresa en la ventana correspondiente.
4. Se acepta el registro del nuevo usuario.
5. Se notifica al usuario que ha sido dado de alta en el sistema.
6. Inicio de sesión en la aplicación.

* Realizar una compra

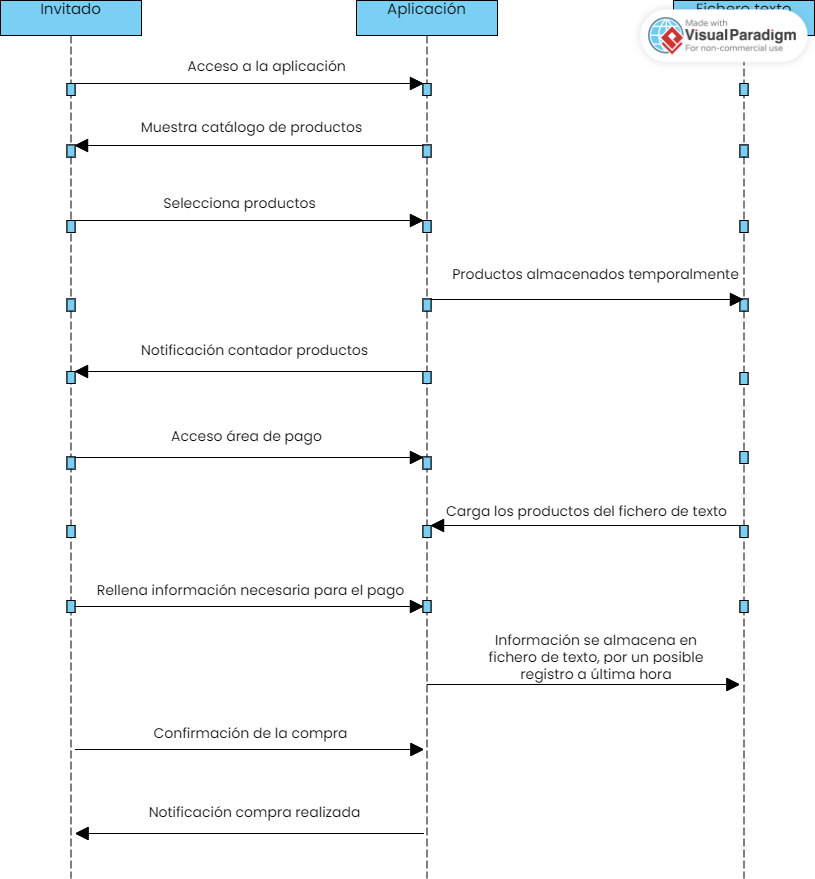


1. El usuario selecciona los productos que desea comprar.
2. Los productos seleccionados se almacenan en el carrito. El contador del carrito aumenta, el cual será visible para el usuario.
3. Accede al área de confirmar la compra.
4. Se introducen los datos necesarios para poder realizar el pago.
5. El sistema comprobará que los datos introducidos son correctos.
6. Se realiza el cobro del total de la compra.
7. Se notifica el cobro.
8. Los productos que se han comprado se almacenan en una tabla del sistema ‘Detalle pedido'. Cada detalle de pedido pertenece a un pedido.
9. Una vez realizada y confirmada la compra, se almacena en el sistema la factura.

Por lo tanto, se le dará la posibilidad de guardar o enviar la factura a su correo.

##### 4.2.2.4.2.- Usuario invitado

* Realizar compra



El usuario invitado tendrá acceso a todas las ventanas principales para la realización de compra, menos aquellas que requieran estar dado de alta en el sistema.

El proceso de compra es similar al de usuario (dado de alta), sin embargo, todos los datos, tanto los productos seleccionados como los datos para el envío, serán almacenados temporalmente en ficheros de texto que están ocultos. Una vez confirmada y realizada la compra, estos ficheros de texto serán eliminados.

Los datos son almacenados temporalmente en los ficheros de texto, por si en el transcurso de la compra, el usuario invitado se anima a darse de alta en el sistema. En este caso, el usuario solo rellena el registro. Una vez hecho esto, todos los datos que están en los ficheros correspondientes, pasan a estar en el sistema y estos se borran.

Por último, los datos necesarios para una posible reclamación en un tiempo estimado, serán almacenados en el sistema. Después de ese tiempo, los datos son eliminados.

### 4.2.3.- Pruebas del sistema

#### 4.2.3.1.- SQL Server Profile

Esta es una herramienta de monitoreo y diagnóstico que se encuentra disponible en Microsoft SQL Server. Permite capturar y analizar eventos y actividades en tiempo real.

Como permite depurar problemas y errores en consultas, transacciones o eventos específicos, según se ha ido desarrollando el proyecto, se hizo uso de esta herramienta para controlar los datos enviados a base de datos. Esto nos ayuda a comprobar si los datos enviados desde la aplicación a base de datos son los correctos.

No se adjunta pantallazos o código porque lo que nos muestra esta herramienta, son los datos enviados a base de datos. Por lo que sí un dato enviado de la aplicación a base de datos no lo hace según lo establecido, la columna de la fila correspondiente a la tabla aparecerá en blanco, en caso contrario, mostrará los datos.

#### 4.2.3.2.- MSTest

En este apartado, se hace uso de pruebas unitarias. Estas pruebas permiten evaluar y verificar el funcionamiento individual y aislado de componentes de código, así asegurar que cada unidad de código se comporte de acuerdo con las expectativas establecidas.

Para las pruebas se hace uso de la herramienta de desarrollo de software llamada MSTest.

MSTest, también conocido como Microsoft Unit Testing Framework, es marco de pruebas unitarias proporcionado por Microsoft para el desarrollo de software en entorno basados en tecnología de Microsoft. Este proporciona una variedad de características y funcionalidades que facilitan la creación y ejecución de pruebas, así como la obtención de resultados claros y precisos.

Para la realización de pruebas sobre un método, se coloca en la parte superior del método, el atributo TestMethod. Si al ejecutar la prueba, el resultado no es igual al resultado esperado, la prueba del método tendrá una verificación en rojo, caso contrario, en verde.

Las pruebas que se mostrarán a continuación, es cómo debería comportarse la aplicación.

Los métodos hacen llamada al procedimiento almacenado correspondiente y en cada una validará los datos. Si las validaciones no son válidas, muestra un mensaje informativo y el proceso dejará de funcionar. Todas estas validaciones están en base de datos.

Las pruebas unitarias, como ya se ha mencionado anteriormente, se realizan de forma individual, por lo que para estas hay que hacerlas individualmente y en el orden establecido.

* Alta usuario

[TestMethod]

public void UsuarioDadoAlta()

{

Cliente cliente = new Cliente("Prueba", "Prueba", 23, "23/08/1999", "Masculino", "antajc23@gmail.com");

Usuario usuario = new Usuario("prueba", "1234", 0);

bool isAlta = BDUsuario.insertarDatos(cliente, usuario);

Assert.AreEqual(true, isAlta);

}

Esta prueba unitaria comprueba si el usuario se ha dado de alta correctamente. Dará de alta a un nuevo usuario. El resultado esperado es true.

* Actualizar datos usuario

[TestMethod]

public void ActualizarDatosUsuario()

{

bool actualizado = BDUsuario.actualiarDatos("prueba", "12345", "correoprueba@gmail.com");

Assert.AreEqual(true, actualizado);

}

Se comprobará si se actualiza el usuario de forma correcta. Se le pasa como parámetro los nuevos datos.El resultado esperado es true.

* Agregar un producto al carrito

[TestMethod]

public void AgregarProductoToCarrito()

{

int idUsuario = BDUsuario.consultarUsuarioName("prueba");

Usuario usuario = new Usuario(idUsuario);

Producto producto = new Producto(1);

Carrito carrito = new Carrito(1, producto);

bool insertado = carrito.insertarProducto(usuario, carrito, false);

Assert.AreEqual(true, insertado);

}

Se agrega un nuevo producto al carrito referenciando al usuario y al producto. Los datos se pasan como parámetros.El resultado esperado es true.

* Eliminar producto del carrito

[TestMethod]

public void EliminarProductoFromCarrito()

{

int idUsuario = BDUsuario.consultarUsuarioName("prueba");

Carrito carrito = new Carrito();

carrito.vaciarCarrito(idUsuario, 1, false);

}

Se elimina un producto del carrito. Los parámetros son el id del usuario, el id del producto. El último parámetro es un booleano (true vaciar completamente el carrito o false borrar un producto). El resultado esperado es true.

* Agregar dirección usuario

[TestMethod]

public void AgregarDireccionUsuario()

{

int idUsuario = BDUsuario.consultarUsuarioName("prueba");

Direccion direccion = new Direccion("Casa", "C/Capricornio, 1", "España", "Zaragoza", "Zaragoza", "50012", "600123321");

bool actualizado = BDDireccion.actualizarDireccion(direccion, idUsuario);

Assert.AreEqual(true, actualizado);

}

Se agrega la dirección referenciando al usuario. Parámetros: el objeto dirección y el id del usuario. El resultado esperado es true.

* Realizar compra

[TestMethod]

public void realizarCompra()

{

int idUsuario = BDUsuario.consultarUsuarioName("prueba");

bool actualizado = BDPedido.actualizarPedido(idUsuario, (int)EMetodoPago.Efectivo);

Assert.AreEqual(true, actualizado);

}

En esta prueba, se realizará la compra. Los parámetros son el id del usuario y el método de pago.Como ejemplo el parámetro método de pago toma el valor efectivo.Con el id del usuario se accederá a todos los datos necesarios para realizar la compra. El resultado esperado es true.

* Dar baja al usuario mediante nombre de usuario y contraseña

[TestMethod]

public void DarBajaUsuarioUsandoUserNameYContrasenia()

{

bool isBaja = BDUsuario.darBajaUsuarioPorUsuario("prueba", "12345");

Assert.AreEqual(true, isBaja);

}

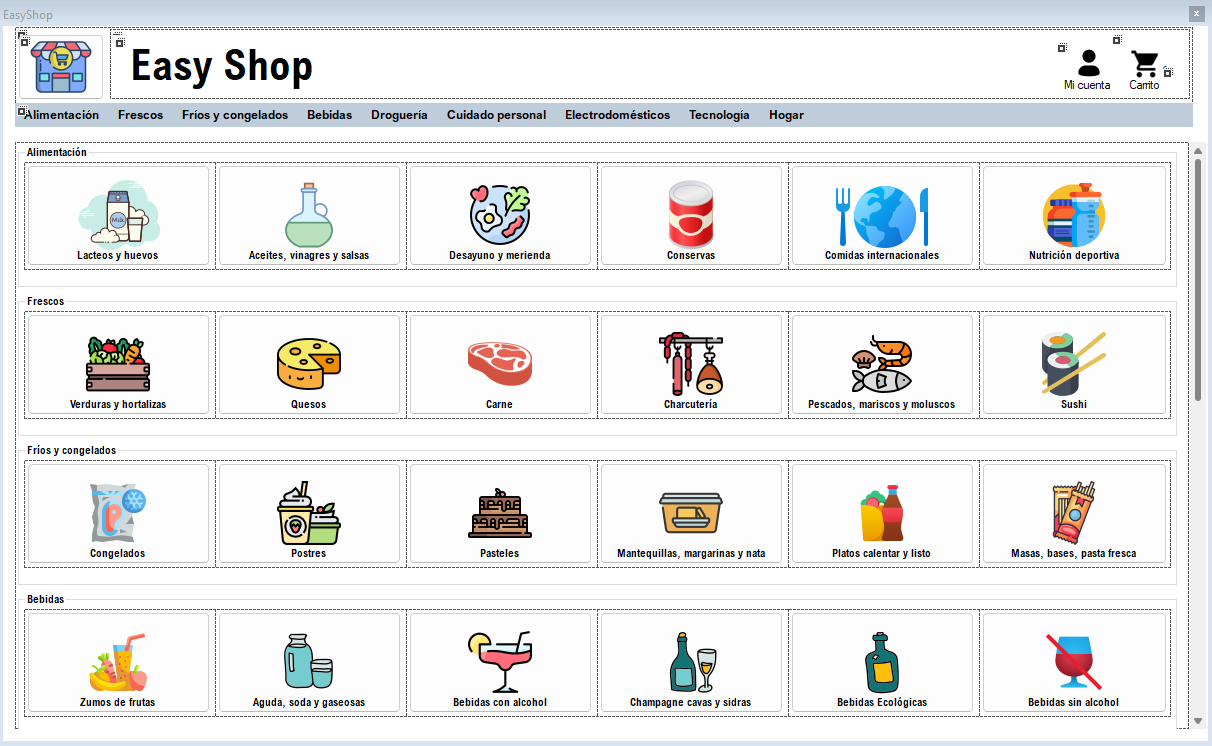
La baja del usuario se hace mediante el nombre del usuario y la contraseña. El resultado esperado es true.

### 4.2.4.- Documentación

##### 4.2.4.1.- Manual de usuario

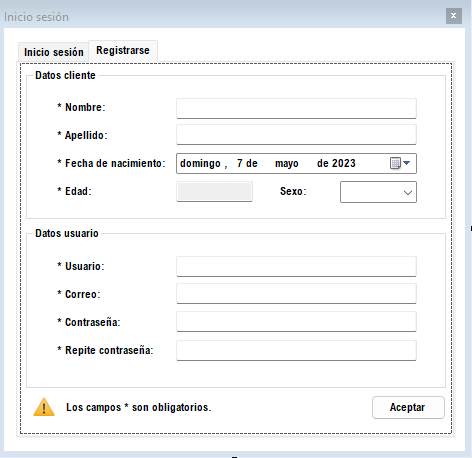
Como toda aplicación es necesario un manual de usuario ya que proporcionará información detallada de cómo utilizar el producto o aplicación. Su objetivo principal es ayudar al consumidor a comprender el funcionamiento del producto y el uso correcto y eficiente de este. Por lo tanto, se proporcionará a continuación información detallada de la aplicación para su uso:

###### 4.2.4.1.1.- Catálogo de categorías y subcategorías



Al iniciar la aplicación, se mostrará el catálogo de productos. Esta ventana está formada por categorías y subcategorías. Desde esta ventana se puede acceder al área de inicio de sesión, registro, perfil, carrito. Además de poder acceder a los productos correspondientes a las categorías.

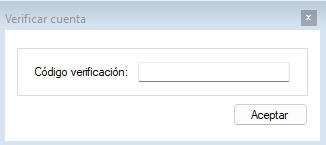
###### 4.2.4.1.2.- Registro



Se rellenan todos los datos obligatorios para darse de alta en el sistema. En caso de que se de el botón Aceptar, y algún dato falte por rellenar, se mostrará en la parte inferior izquierda un mensaje de error.

Si se han rellenado todos los campos, se comprobará que tanto el nombre de usuario como el correo no estén en uso, si es así se notificará.

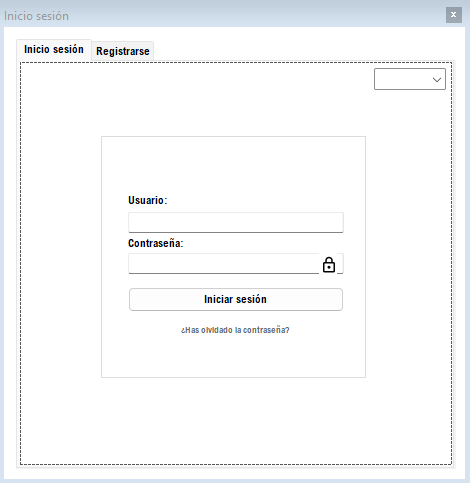
###### 4.2.4.1.3.- Verificar cuenta



Una vez rellenado todos los campos obligatorios y estos estén verificados, se enviará un mensaje al correo electrónico proporcionado con el código de verificación. Si el código introducido en el campo correspondiente no coincide con el enviado al correo, se mostrará un mensaje advirtiendo del error, además de dar la posibilidad de enviar un código nuevo.

Si el código coincide, se le da de alta en el sistema.

###### 4.2.4.1.4.- Inicio sesión

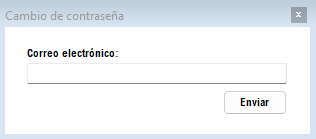


Una vez finalizado el registro, se muestra la ventana para poder iniciar sesión.

En caso de que algún campo falte por rellenar o no se pueda iniciar sesión, se le notificará.

Desde esta misma, se puede recuperar la contraseña.

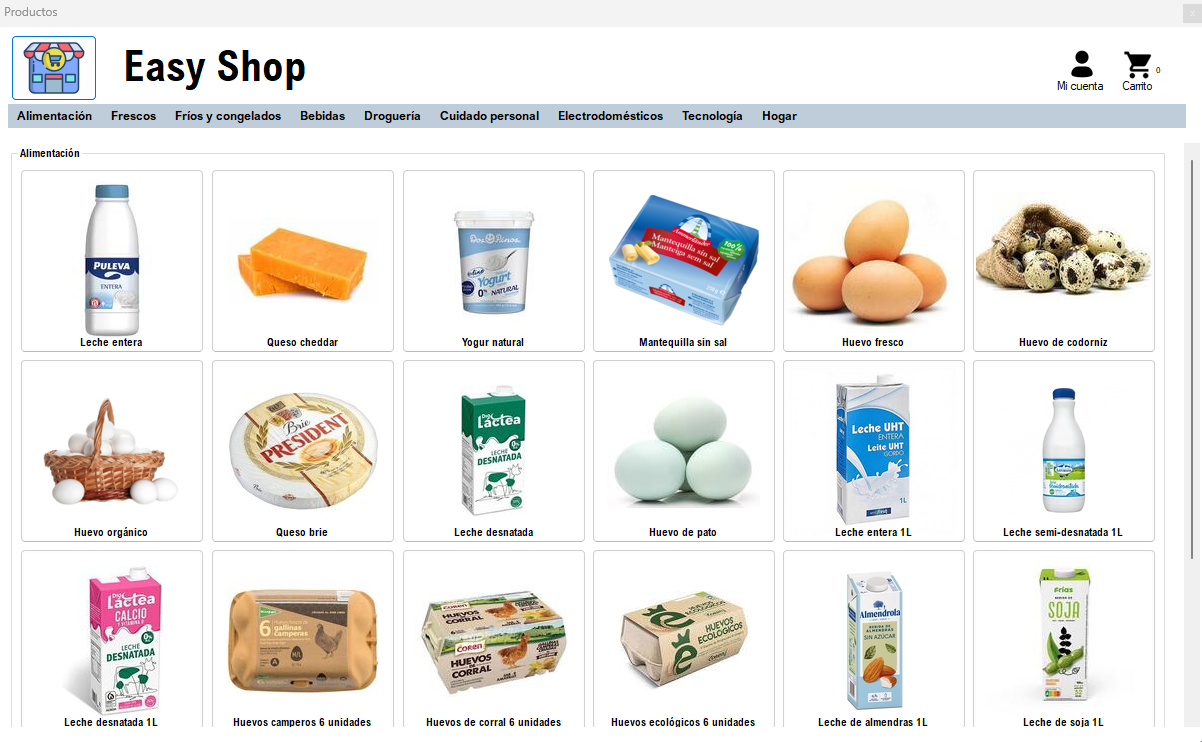
###### 4.2.4.1.5.- Cambio de contraseña



En caso de que se desee recuperar la contraseña, se enviará un código alfanumérico al correo proporcionado en el registro de la cuenta.

El código enviado será la contraseña con la que se podrá acceder a la cuenta. Una vez iniciada sesión, se recomienda cambiar la contraseña.

###### 4.2.4.1.6.- Selección del producto



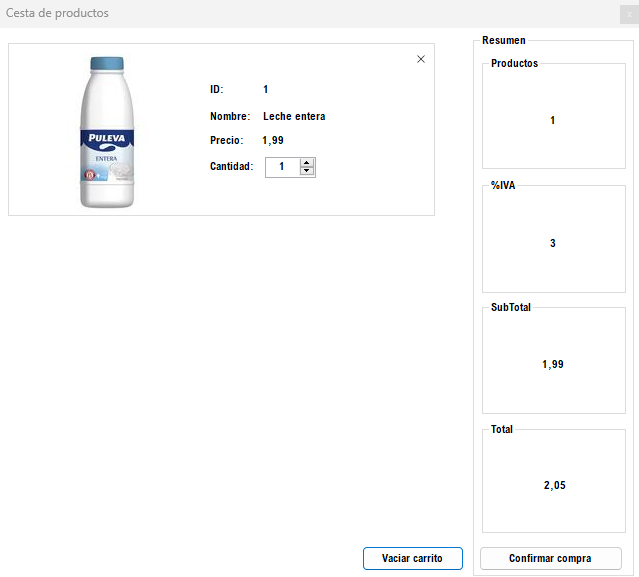
Se accede desde la [ventana principal](#_ocvuxikienco). Dentro de una subcategoría se muestran un catálogo de productos correspondiente a la subcategoría.

###### 4.2.4.1.7.- Agregar producto al carrito de productos



Al seleccionar un producto del catálogo de productos de la subcategoría, se muestra la ventana Detalle del producto con detalles del producto. Desde esta misma se puede modificar la cantidad del producto y añadirla al carrito de productos.

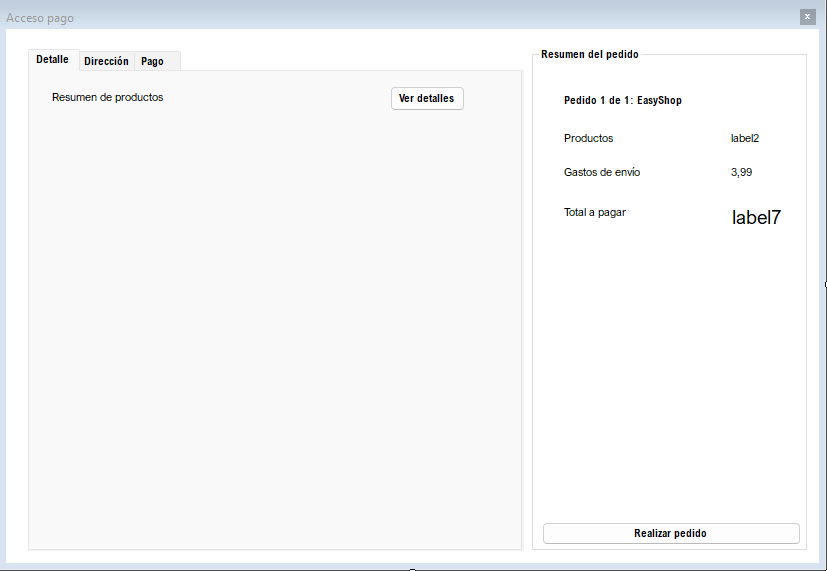
###### 4.2.4.1.8.- Carrito de productos



Se accede desde el catálogo de las categorías o subcategorías y desde los productos correspondientes. Desde esta ventana se podrán visualizar todos los productos añadidos al carrito, además de poder modificar la cantidad del producto, se podrá eliminar un producto del carrito o simplemente vaciarlo o confirmar la compra.

Por último, se muestra un resumen breve del producto añadido y un resumen de lo que sería la compra de los productos del carrito.

###### 4.2.4.1.9.- Confirmación de la compra



Tanto el usuario registrado en el sistema como el usuario invitado (no dado de alta en el sistema) tendrán acceso a esta ventana para confirmar la compra.

* Usuario:

En esta ventana, se recogen los productos del carrito de la base de datos y se calcula el total a pagar. Una vez elegido el método de pago y la dirección en sus pestañas correspondientes, el botón ‘Realizar pedido’ se activará.

El usuario al confirmar la compra, se almacenan los productos comprados en una tabla llamada ‘linea\_pedido’, esta forma parte de un pedido y este formará parte de la factura. Todas las compras realizadas serán almacenadas y podrán ser consultadas en la aplicación.

* Invitado:

Los productos están en un fichero de texto que representa al carrito de productos. Todos estos son recogidos y se calcula el total a pagar.

Una vez que se elija el método de pago y la dirección en sus pestañas correspondientes, el botón ‘Realizar pedido’ se activará.

Se dan de alta los datos que se han rellenado en el momento. En un tiempo determinado, esos datos serán borrados.

###### 4.2.4.1.10.- Perfil



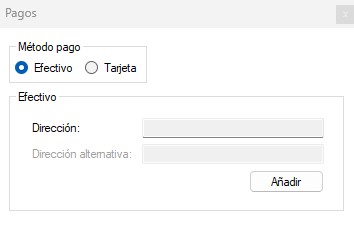
Se accede desde la ventana principal. Además de mostrar todos los datos del usuario, se permite acceder a otra ventana para modificar estos mismos. Por último, se permite acceder a la ventana ‘Otras opciones’ y cerrar la sesión.

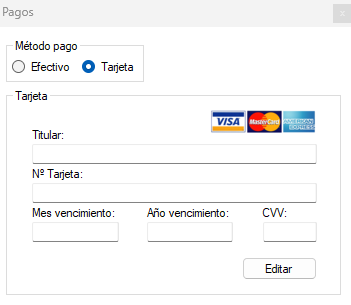
###### 4.2.4.1.11.- Otras opciones



Se accede desde la ventana Perfil. Desde esta misma se tiene acceso a las siguientes ventanas: Pagos, Pedidos, Direcciones y Darse de baja.

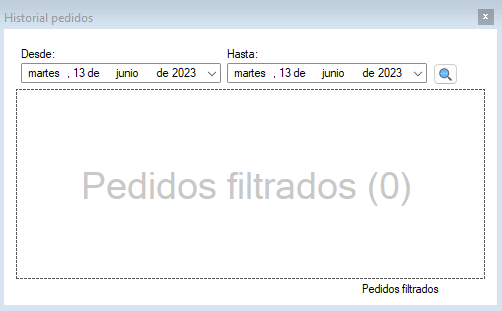
###### 4.2.4.1.12.- Pagos





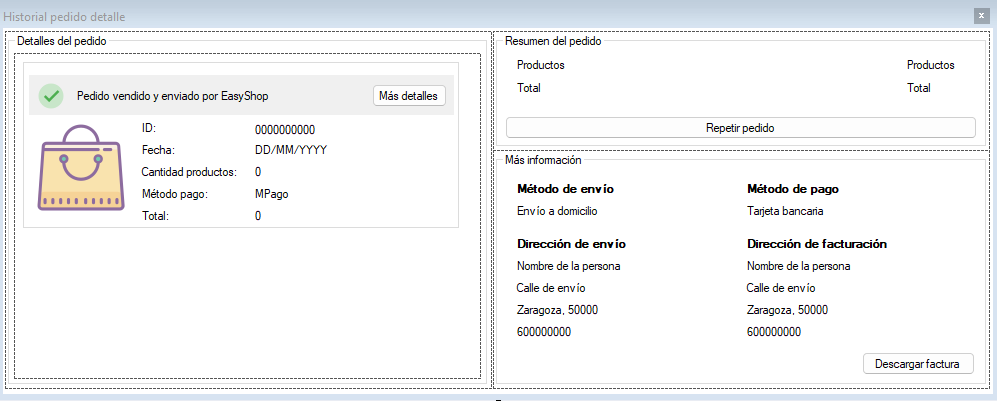
Esta ventana mostrará los datos proporcionados a la hora de confirmar la compra. Aunque desde esta misma, se puede agregar la tarjeta de crédito. Esta ventana, además de acceder desde el perfil, se mostrará al confirmar la compra.

###### 4.2.4.1.13.- Pedidos



Se muestran los pedidos filtrados por fecha de compra. Más adelante se permitirá filtrar por el Id de la factura. Al mostrar los pedidos, se puede acceder a la ventana Historial pedido detalle.

###### 4.2.4.1.14.- Más detalles

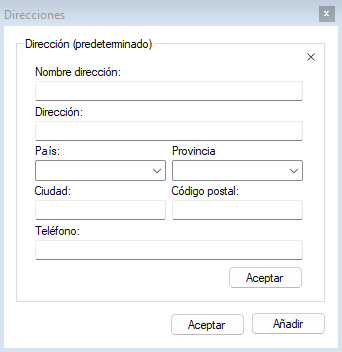


En esta ventana se dará un detalle más extenso de la compra realizada en la fecha.

Desde esta misma podrá añadir el producto que compró en su momento a su carrito de productos o incluso repetir el pedido completo.

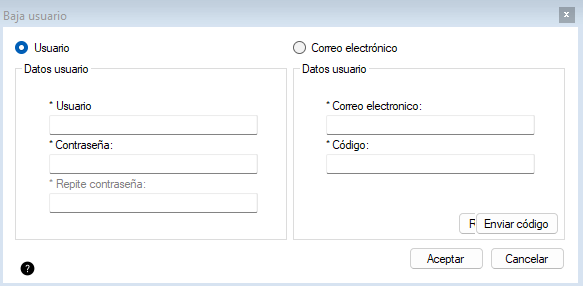
Además, desde esta ventana puede descargar la factura del pedido.

###### 4.2.4.1.15.- Direcciones



Muestra la dirección proporcionada al confirmar la compra. Aunque también puedes agregar dirección/es desde esta misma. Además se podrá editar la dirección e incluso añadir una dirección como alternativa.

###### 4.2.4.1.16.- Darse de baja



Para darse de baja se proporciona dos formas de hacerlo:

1. Mediante el usuario

Se rellenan todos los datos necesarios para poder realizar la operación ‘Dar de baja’. Si tanto el usuario como la contraseña son erróneas o no coincide su verificación, se mostrará un mensaje informando de dicho error.

1. Mediante el correo electrónico

Se envía un código al correo proporcionado. Este código será introducido en su campo correspondiente. Si este coincide, se realizará la operación ‘Dar de baja’. En caso de que el código no corresponda, se notificará. Por último, se da la posibilidad de reenviar el código.

**Punto a destacar:**

Cabe destacar que el usuario invitado tendrá acceso a las mismas ventanas para realizar la compra, excepto aquellas que requieran estar dado de alta en el sistema, como lo es la ventana Perfil o como la ventana Pedidos etc.

En modo usuario invitado,los productos del carrito en vez de almacenarse temporalmente hasta que se confirme la compra en la base de datos (modo usuario), estos se alojarán en un fichero de texto, así mismo con todos los datos necesarios para realizar la compra de modo invitado.

##### 4.2.4.2.- Manual de instalación o configuración

El manual de instalación o configuración proporcionará instrucciones básicas para configurar la aplicación.

###### 4.2.4.2.1.- Usuario final

1. Requisitos del sistema:
   1. Sistema operativo: Windows 10 o posterior. Próximamente para otros sistemas operativos.
   2. Memoria RAM: Mínimo 4 GB.
   3. Espacio en disco: Mínimo 500 MB de espacio disponible.
2. Descarga e instalación:
   1. Obtén el ejecutable de instalación desde un medio autorizado.
   2. Ejecuta el archivo de instalación y sigue las instrucciones para completar el proceso de instalación. Selecciona una ubicación de instalación adecuada.
3. Configuración inicial:
   1. Al finalizar la instalación, ejecuta la aplicación por primera vez.
   2. Aparecerá un asistente de configuración inicial que te guiará a través de los pasos básicos para establecer la configuración de la aplicación.
4. Idioma y preferencias:
   1. Selecciona el idioma. Por ahora solo está disponible para España.
5. Crea cuenta de usuario:
   1. Date de alta en el sistema para poder acceder a la aplicación y poder tener futuros descuentos y ofertas.

###### 4.2.4.2.2.- Desarrollador

1. Base de datos

Como se ha mencionado anteriormente, la base de datos empleada para el desarrollo del proyecto es SQL Server. Las consultas se hacen mediante procedimientos almacenados que están adjuntadas en el proyecto en la carpeta llamada ‘Procedimientos almacenados’.

Para verificar los datos enviados desde la aplicación a base de datos se emplea la herramienta incorporada en SQL Server, llamada SQL Profile. En caso de que el proyecto lo esté desarrollando más de un programador, se recomienda filtrar el nombre del PC para recibir solamente los datos enviados que les corresponde.

**¡Ojo!**

Los procedimientos almacenados se deben ejecutar en SQL Server, si no dará error.

Cambiar la ruta del servidor y de la base de datos (si esta última llega a cambiarse), ya que la base de datos es local, no es remoto. Por lo tanto, al cambiar de PC, la ruta del servidor varía. La ruta de la base de datos se encuentra en la carpeta Clases/Servidor.

1. Entorno de desarrollo

El entorno de desarrollo es Visual Studio 2022.

1. Lenguaje de programación

El proyecto está desarrollado en C#.

1. Reportes

Hay que descargar una extensión llamada ‘Microsoft RDLC Report Designer 2022’. Si esta extensión no está descargada en el entorno de desarrollo Visual Studio, el proyecto tendrá errores de compilación.

Una vez descargada esta extensión, hay que agregar el origen de datos desde Herramientas.

1. Pruebas unitarias

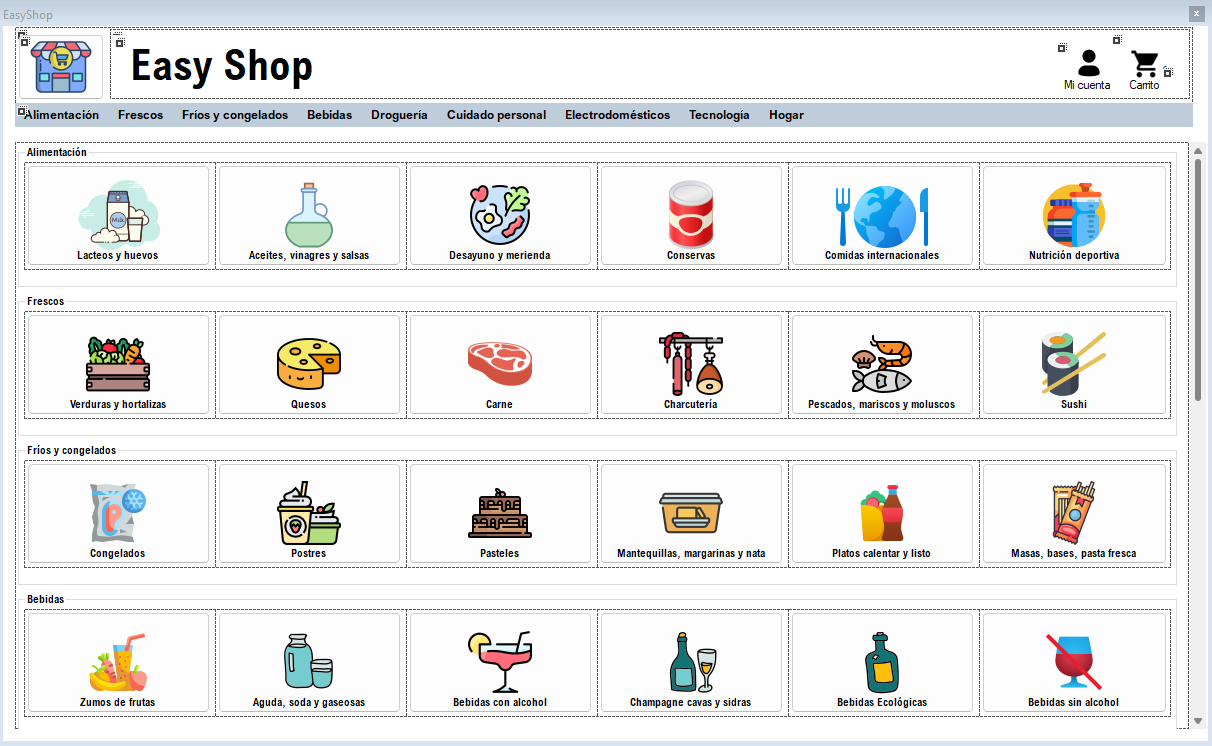
Para las pruebas unitarias, se agregó un nuevo proyecto del tipo MSTest.

## 4.3.- Diseño de interfaz gráfica

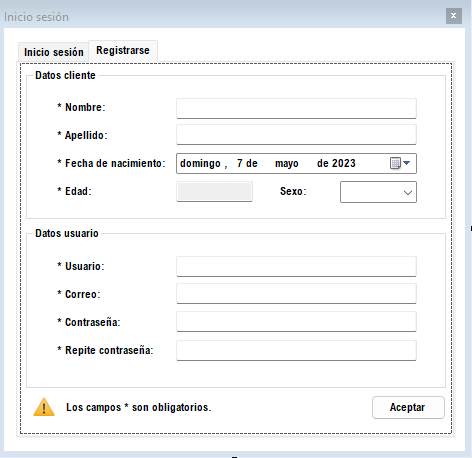
En este apartado, solamente se muestran las ventanas gráficas usadas en la aplicación.

La explicación más detallada se hace en el apartado anterior a este.

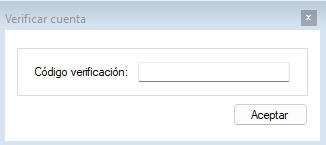
### 4.3.1.- Catálogo de categorías y subcategorías



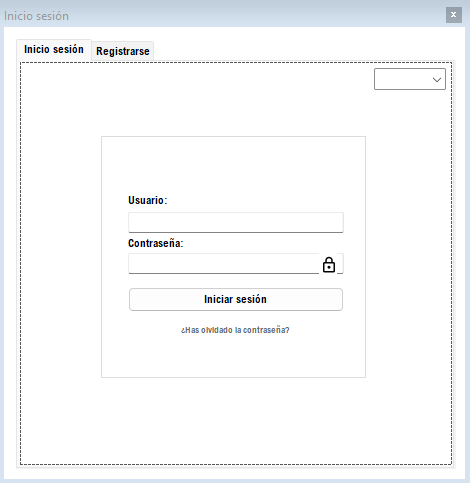
### 4.3.2.- Registro



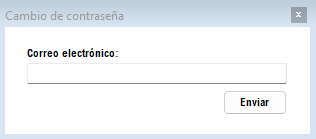
### 4.3.3.- Verificar cuenta



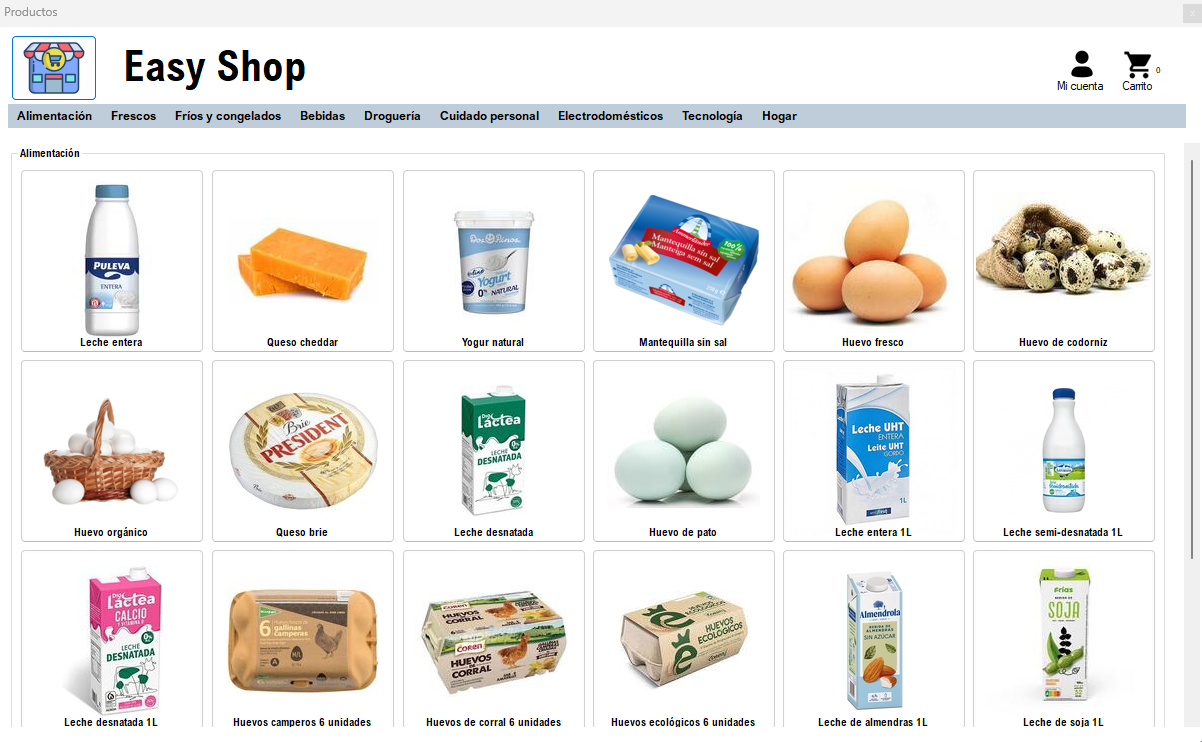
### 4.3.4.- Inicio sesión



### 4.3.5.- Recuperar contraseña



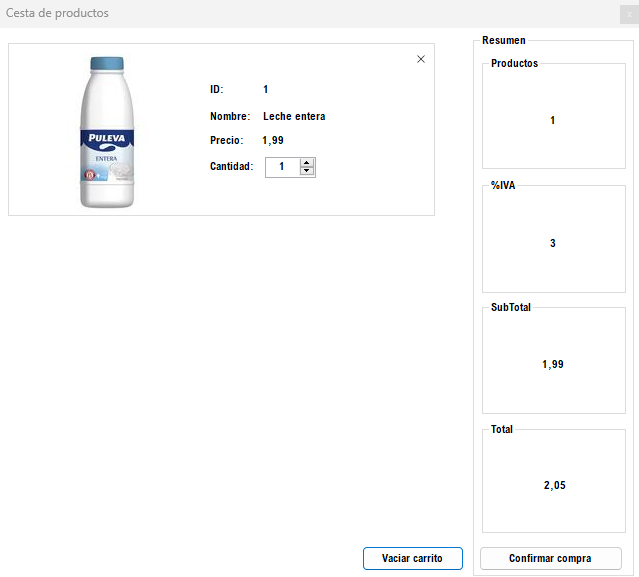
### 4.3.6.- Selección del producto



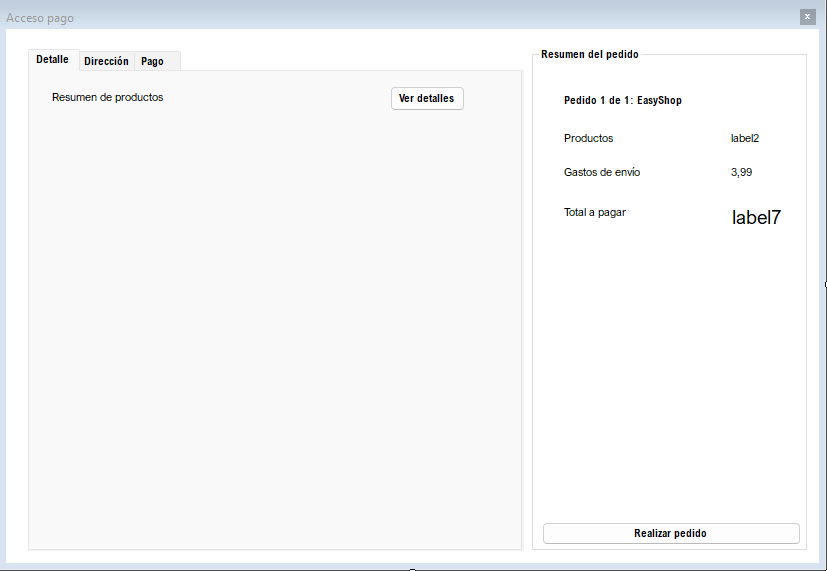
### 4.3.7.- Agregar producto al carrito de productos



### 4.3.8.- Carrito de productos



### 4.3.9.- Confirmación de la compra



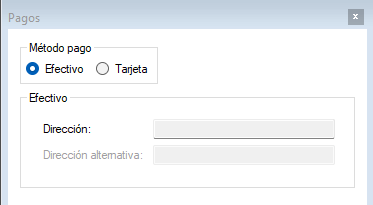
### 4.3.10.- Perfil

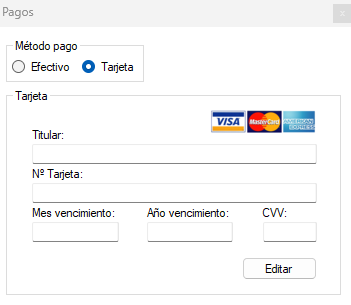


### 4.3.11.- Otras opciones

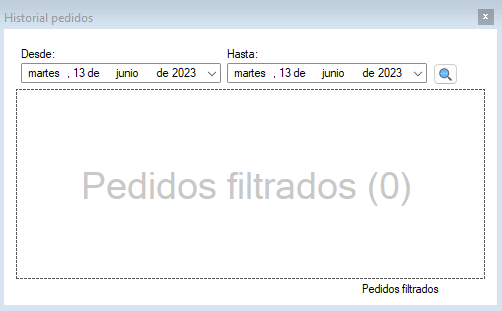


### 4.3.12.- Pagos

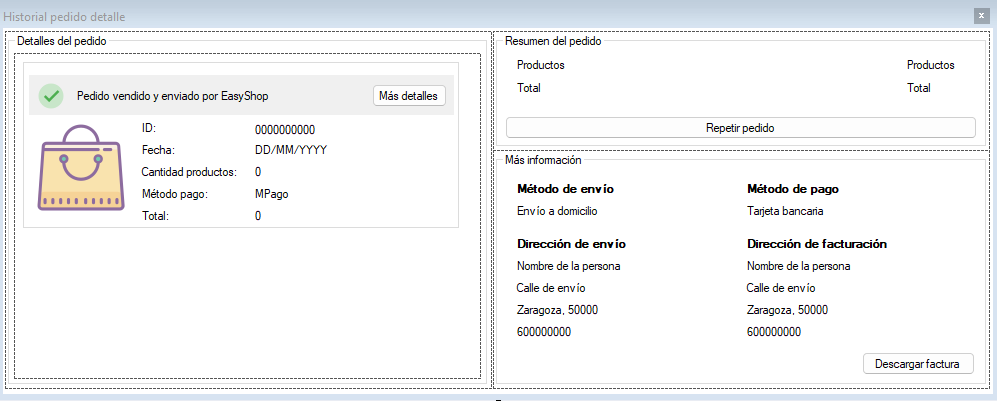




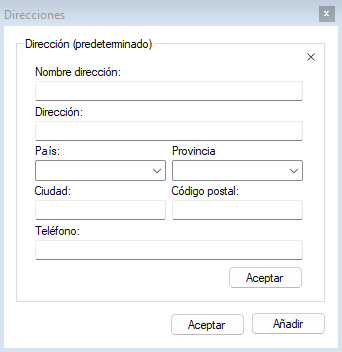
### 4.3.13.- Pedidos



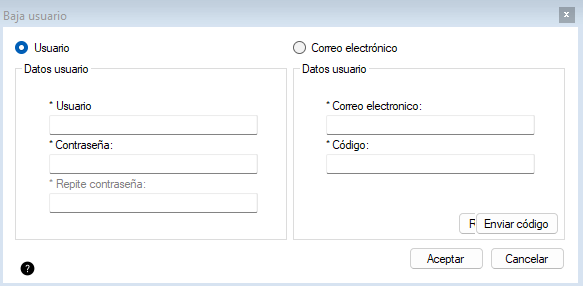
### 4.3.14.- Más detalles



### 4.3.15.- Direcciones



### 4.3.16.- Darse de baja

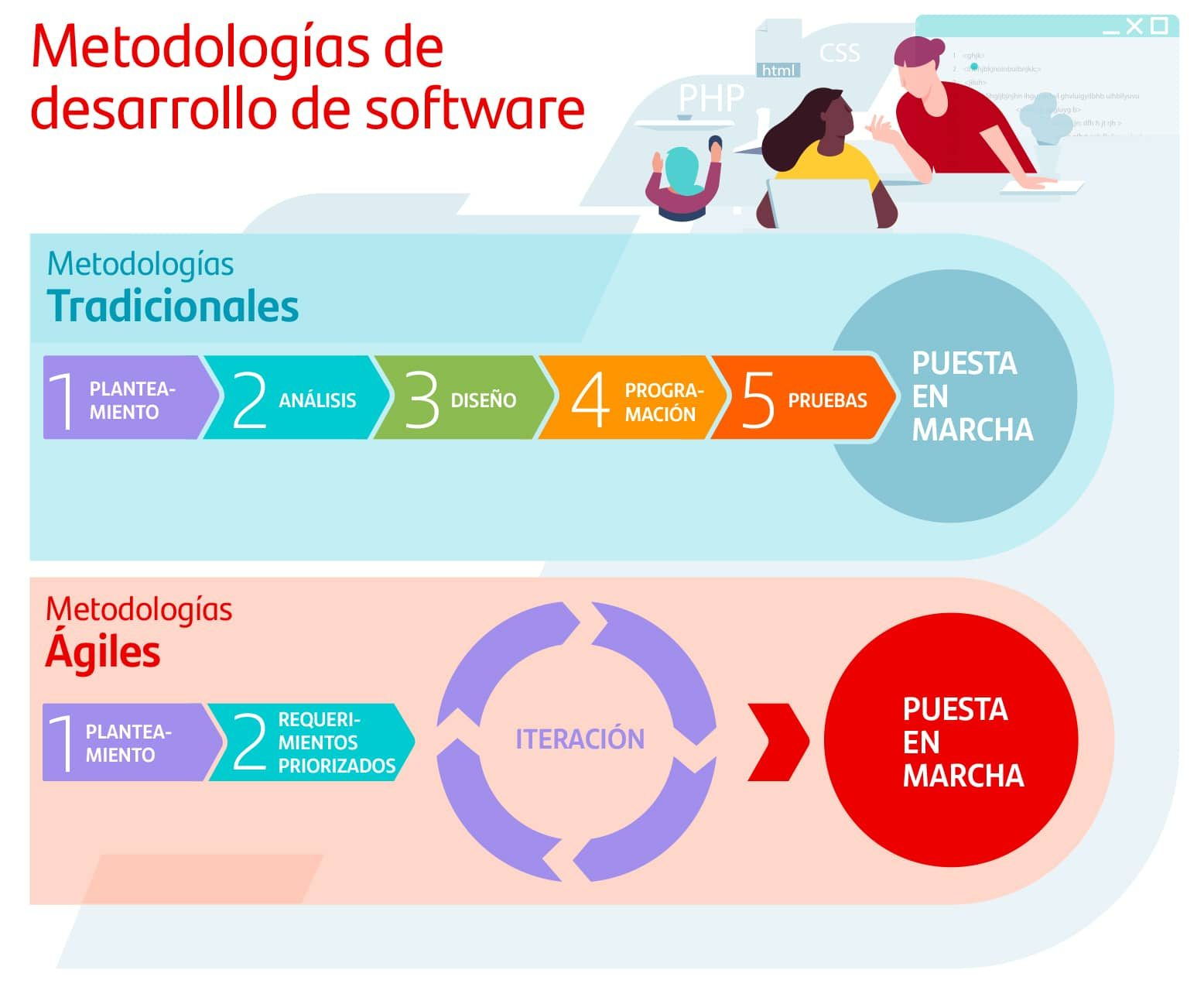


## 4.4.- Metodología seguida

Para el desarrollo de la aplicación, se hace uso de la metodología de desarrollo de software tradicional llamada Waterfall (Cascada).

Las diferentes funciones son desarrolladas en etapas diferenciadas y obedeciendo un riguroso orden. Antes de pasar de etapa, debe estar finalizada la anterior. Por lo tanto, los requisitos y especificaciones iniciales no están predispuestos para cambiarse y los resultados no se pueden apreciar hasta que el proyecto no esté bastante avanzado.

A continuación se muestra un esquema que explica de forma gráfica en qué consiste este tipo de metodología:



Este esquema ha sido explicado en el punto [4.2.1](#_rdprc1amvsdl)

## 4.5.- Recursos

### 4.5.1.- Materiales

Para el desarrollo de esta aplicación, se ha requerido de software de desarrollo en C# y SQL Server para la base de datos. Para comprobar los datos enviados desde la aplicación a la base de datos, se ha hecho uso de la herramienta Sql Server Profiler.

### 4.5.2.- Infraestructuras y equipamiento

Se necesitó un espacio de trabajo cómodo, con buena iluminación, teclado, ratón, así como una pantalla adecuada para poder visualizar la interfaz de usuario de la aplicación.

### 4.5.2.- Humanos

Para el desarrollo de la aplicación, se requirió de una persona para llevar a cabo todas las etapas de la metodología de desarrollo de software. Además de una investigación con tiempo extra del planificado para ciertos puntos en el transcurso del desarrollo.

# 5.- Presupuesto

El presupuesto que se presentará a continuación estimará los costos involucrados en la creación de la aplicación para 40 horas trabajadas. Este presupuesto sigue la metodología elegida para el desarrollo de la aplicación (En cascada):

1. Planteamiento: 13h x 70€/h = 910€
   1. Reuniones iniciales con el cliente para entender el objetivo, alcances y requisitos.
   2. Análisis de mercado y competencia para identificar oportunidades y características clave.
2. Análisis: 7h x 70€/h = 490€
   1. Investigación y recopilación de requisitos detallados de la aplicación, incluyendo funcionalidades, flujos de trabajo y usuarios objetivo.
   2. Creación de todo tipo de diagramas para visualizar la interacción entre los diferentes componentes.
   3. Estimación del esfuerzo y los recursos necesarios para el desarrollo.
3. Diseño: 8h x 70€/h = 560€
   1. Diseño de la arquitectura de la aplicación, incluyendo la estructura de la base de datos, la lógica de negocio y la interfaz de usuario.
   2. Definición de colores, estilos visuales y las pautas de diseño.
4. Programación: 7h x 70€/h = 490€
   1. Desarrollo de la aplicación utilizando tecnologías específicas, como lo son los lenguajes de programación y frameworks.
   2. Implementación de las funcionalidades principales, como la gestión de productos, carrito de compra, los pagos y la integración de servicios externos.
   3. Optimización del rendimiento y la seguridad de la aplicación.
5. Pruebas: 2.5h x 70€/h = 175€
   1. Realización de pruebas unitarias para verificar el correcto funcionamiento de la aplicación.
   2. Pruebas de usabilidad y experiencia de usuario para garantizar una interfaz intuitiva y fácil de usar.
   3. Identificación y corrección de errores o fallos detectados durante las pruebas.
6. Puesta en marcha: 2.5h x 70€/h = 175€
   1. Configuración del entorno de producción y preparación de la aplicación para el lanzamiento.
   2. Implementación de medidas de seguridad y protección de datos.

Total estimado del presupuesto: **2.800€**

Si se trabajase más horas de las establecidas en el presupuesto inicial, por cada hora adicional en cada parte del desarrollo del proyecto, implica calcular el costo de las horas adicionales y sumarlo al presupuesto total.

# 6.- Resultados objetivos y conclusiones

El resultado objetivo de este proyecto es el desarrollo de una aplicación que cumpla con las necesidades y expectativas de los usuarios. Se busca crear una aplicación funcional, intuitiva y eficiente, que proporcione una experiencia de compra en línea satisfactoria.

El objetivo principal es ofrecer a los usuarios la posibilidad de buscar, visualizar y comprar productos de manera fácil y conveniente, a través de una interfaz amigable y con opciones de pago seguras. La integración exitosa de servicios externos, como sistemas de pago y servicios de entrega, es un componente clave para garantizar una experiencia de compra completa.

En conclusión, se busca brindar a los usuarios una solución efectiva y agradable para realizar sus compras en línea.

# 7.- Bibliografía

[Envío de correo](https://learn.microsoft.com/es-es/visualstudio/vsto/how-to-programmatically-send-e-mail-programmatically?view=vs-2022)

[Controles personalizados](https://learn.microsoft.com/es-es/dotnet/desktop/winforms/controls/developing-custom-windows-forms-controls?view=netframeworkdesktop-4.8)

[Procedimientos almacenados](https://learn.microsoft.com/en-us/aspnet/web-forms/overview/data-access/advanced-data-access-scenarios/creating-stored-procedures-and-user-defined-functions-with-managed-code-cs)

[Reportes](https://learn.microsoft.com/en-us/sql/reporting-services/application-integration/using-the-winforms-reportviewer-control?view=sql-server-ver16)

[Pila Stack](https://learn.microsoft.com/es-es/dotnet/api/system.collections.generic.stack-1?view=net-7.0)

[Enumerados](https://learn.microsoft.com/es-es/dotnet/csharp/language-reference/language-specification/enums)

[Ficheros de texto](https://learn.microsoft.com/es-es/troubleshoot/developer/visualstudio/csharp/language-compilers/read-write-text-file)

Temario del ciclo