

# I/O RECORDS

Especificación de Requisitos Software



## Autores

David Arroyo Segovia  
Ignacio Cepeda Bajo  
Ángel Cruz Alonso  
Hao Hao He  
Carla Peñarrieta Uribe

**Control de cambios**

<b>Número de versión</b>	<b>Fecha</b>	<b>Autores</b>	<b>Descripción</b>
1	15/12/2016	Ángel Cruz David Arroyo Hao Hao He Carla Peñarrieta	Comienzo del trabajo del: 1. Introducción y 2. Descripción general.
2	16/12/2016	David Arroyo Ángel Cruz	Ampliación y revisión de la Introducción y de la Descripción general.
3	17/12/2016	David Arroyo Ángel Cruz	Estudio de las posibles funciones de la aplicación. Listado de las funciones y módulos.
4	18/12/2016	Ignacio Cepeda Ángel Cruz	Redacción apartado 3
5	20/12/2016	Ignacio Cepeda	Revisión y ampliación de los demás apartados
6	29/12/2016	Carla Peñarrieta	Realización de los diagramas de casos de uso
7	30/12/2016	Hao Hao He	Funciones para los diagramas de funciones de requisitos
8	30/12/2016	Carla Peñarrieta	Realización de algunos diagramas de actividad
9	2/1/2016	Carla Peñarrieta	Finalización de los diagramas de actividad
10	14/01/2017	David Arroyo	Ampliación y corrección de toda la SRS

# Índice

<b>Índice</b>	<b>3</b>
<b>Introducción</b>	<b>4</b>
Propósito	4
Alcance	4
Definiciones, acrónimos y abreviaturas	5
Referencias	6
Resumen	6
<b>Descripción general</b>	<b>6</b>
Perspectiva del producto	6
Funciones del producto	7
Características del usuario	8
Restricciones	8
Supuestos y dependencias	9
Requisitos futuros	8
<b>Requisitos específicos</b>	<b>9</b>
Interfaces externos	9
Requisitos funcionales	10
Requisitos de rendimiento	32
Requisitos sobre la persistencia de datos	32
Restricciones de diseño	34
Atributos del sistema software	35
<b>Apéndices</b>	<b>36</b>

# 1. Introducción

## 1.1. Propósito

El documento de especificación de requisitos de software (SRS) tiene como propósito describir el comportamiento de forma completa el sistema que se va a desarrollar. Esta especificación es un medio de comunicación entre cada una de las personas involucradas en el desarrollo de software y, por eso mismo, está sujeto a revisiones, tanto de los desarrolladores como de los usuarios, hasta que se obtenga su aprobación. En cuanto esto ocurra, el documento servirá de base al equipo de desarrollo para la construcción de la aplicación. Este documento se centrará “en qué hace” nuestro sistema, sin abordar el “cómo lo hace”. Para ello, desarrollará un conjunto de casos de uso también llamados requisitos funcionales. Sin embargo, también contiene requisitos no funcionales.

Este documento está dirigido tanto al cliente como al equipo de desarrollo. A su vez, es una justificación de las decisiones tomadas al desarrollar este proyecto, cara al profesor que lo evalúe.

## 1.2. Alcance

Se adjudica al proyecto el nombre I/O Records. A partir de este momento cualquier referencia a I/O Records corresponde al proyecto que nos ocupa.

La aplicación tiene como objetivo gestionar un negocio de venta de discos de música, aunque también podría ser útil para otro negocio minorista similar. El software realizará tareas de gestión del inventario de productos, de ventas y de clientes. Por tanto, no se trata de una herramienta para gestionar tareas como las compras a proveedores o la gestión del personal.

Los objetivos principales que desea cubrir el software son: facilitar y agilizar la administración del catálogo de la tienda, administrar las ventas de una forma sencilla para que cualquier nuevo empleado pueda incorporarse rápidamente al negocio y que los clientes que acudan a la tienda puedan realizar compras desde la misma aplicación.

La meta propuesta es disponer de una aplicación informática que permita gestionar el inventario, y todo lo relativo a administrar una tienda de música. Es una herramienta que sobre todo ayuda a los trabajadores a realizar sus funciones. El ahorro de tiempo y de recursos beneficiará a la empresa, pudiendo reinvertirlos en mejoras que aumenten progresivamente el tamaño de la empresa.

### 1.3. Definiciones, acrónimos y abreviaturas

**Existencias:** Mercancías destinadas a la venta, guardadas en un almacén o tienda.

**Catálogo:** Relación ordenada en la que se incluyen o describen de forma individual libros, documentos, personas, objetos, etc., que están relacionados entre sí.

**JAVA:** Java es un lenguaje de programación de propósito general, concurrente, orientado a objetos que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible.

**Hardware:** Conjunto de elementos físicos o materiales que constituyen una computadora o un sistema informático.

**Software:** Conjunto de programas y rutinas que permiten a la computadora realizar determinadas tareas.

**Interfaces:** Conexión funcional entre dos sistemas o dispositivos de cualquier tipo, dando una comunicación entre distintos niveles.

**GUI:** La interfaz gráfica de usuario, conocida también como GUI (del inglés graphical user interface), es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

**Base de datos (BD):** Son datos relativos a diversas temáticas y categorizados de distinta manera, pero que comparten entre sí algún tipo de vínculo o relación que busca ordenarlos y clasificarlos en conjunto.

**Ethernet:** Es un estándar de redes de área local para computadores con acceso al medio por detección de la onda portadora y con detección de colisiones (CSMA/CD). Su nombre viene del concepto físico de ether.

**Fibra óptica:** Filamento de material dieléctrico, como el vidrio o los polímeros acrílicos, capaz de conducir y transmitir impulsos luminosos de uno a otro de sus extremos; permite la transmisión de comunicaciones telefónicas, de televisión, etc., a gran velocidad y distancia, sin necesidad de utilizar señales eléctricas.

**Procesador:** Componente electrónico donde se realizan los procesos lógicos.

**Memoria:** Dispositivo de una máquina donde se almacenan datos o instrucciones que posteriormente se pueden utilizar.

**Windows:** Windows es un sistema operativo desarrollado por la empresa de software Microsoft Corporation, el cual se encuentra dotado de una interfaz gráfica de usuario basada en el prototipo de ventanas (su nombre en inglés).

**Linux:** Linux es un sistema operativo de software libre (no es propiedad de ninguna persona o empresa), por ende, no es necesario comprar una licencia para instalarlo y utilizarlo en un equipo informático.

**Mac OS:** Es el nombre del sistema operativo creado por Apple para su línea de computadoras Macintosh. Es conocido por haber sido uno de los primeros sistemas dirigidos al gran público en contar con una interfaz gráfica compuesta por la interacción del mouse con ventanas, iconos y menús.

## 1.4. Referencias

- "Create editable sequence diagram with Rational Software Architect", 19/6/2012,  
<https://www.ibm.com/developerworks/rational/library/create-editable-sequence-diagram/>

- "UML, ejemplo sencillo sobre Modelado de un Proyecto",  
<https://msdn.microsoft.com/es-es/library/bb972214.aspx>

"Definición de Entidad (base de datos)", 14/7/2016,  
<http://www.alegsa.com.ar/Dic/entidad.php>

"Integridad de las Bases de Datos - EcuRed",  
[https://www.ecured.cu/Integridad\\_de\\_las\\_Bases\\_de\\_Datos](https://www.ecured.cu/Integridad_de_las_Bases_de_Datos)

## 1.5. Resumen

A continuación, la especificación de requisitos consistirá en una descripción general del producto y del conjunto de requisitos que estarán aplicados a la hora de diseñarlo. Se tratarán los distintos aspectos de la aplicación, así como de la perspectiva del producto, sus funcionalidades o las características del usuario. Continuaremos con los requisitos específicos y, finalmente, expondremos en el apéndice otras ideas para mejorar la aplicación.

# 2. Descripción general

## 2.1. Perspectiva del producto

El producto que se va a desarrollar está concebido para la gestión de una tienda de discos de música. A la aplicación de escritorio se podrá acceder desde cuentas con privilegios diferentes.

En concreto, habrá una cuenta destinada a los usuarios de la tienda que quieran interactuar con el catálogo para efectuar una compra, y a su vez, el personal de la tienda podrá realizar tareas relativas a la gestión del inventario, clientes, ventas y discos.

El personal de la tienda podrá tener un completo control del inventario del negocio. Pudiendo añadir o quitar productos, generar facturas, ver estadísticas de ventas o establecer ofertas.

Por otro lado, el comprador podrá tener toda la información de los productos en venta, productos en oferta o recomendaciones personales. También tendrá la posibilidad de realizar un pedido pudiéndose comprobar la disponibilidad de las existencias. Y una vez realizado dicho pedido, se establecerá una comunicación dependiente-comprador que tramitará el pedido, obtendrá el precio final y, una vez confirmado, se modificarán las existencias actuales.

El objetivo de esta aplicación es dinamizar la comunicación empleado-comprador, así como la completa gestión de las existencias actuales. También la posibilidad de dotar al usuario de una perfecta información sobre los productos.

Las interfaces que tendrá esta aplicación serán dos. Por parte del cliente, tendrá la información del producto, efectuar una compra, devolverla y gestionar su cuenta. En cambio, el dependiente podrá realizar cambios en el catálogo, ver los beneficios, y además podrá hacer uso de todas las funciones que tiene disponibles un cliente.

## 2.2. Funciones del producto

Las funciones principales de nuestra aplicación son las siguientes:

**Gestión de una base de datos:** El personal tendrá la posibilidad de modificar cualquier elemento de la base de datos, como los discos, los pedidos y las cuentas de usuario. El usuario sólo tendrá acceso y podrá modificar su cuenta de usuario y sus pedidos.

Para los pedidos podremos modificar el número de pedidos, los productos, la cantidad de productos y la forma de recogida del producto.

La sección de los datos personales del usuario tendrá acceso al nombre del usuario, la contraseña, los datos y el tipo de usuario que se ha registrado (personal o cliente).

**Adquirir/Devolver - Alta/Baja:** Estas funciones están relacionadas con el registro de datos, ya sean discos, pedidos o usuarios. Esta funcionalidad recoge la gran mayoría de funciones que aparecerán implementadas, como son:

- Añadir / devolver disco
- Dar de alta / de baja usuario
- Añadir / eliminar pedido
- Modificar discos, usuarios, pedidos

**Ver estadísticas:** Recogerá la información de la BBDD, haciendo un análisis absoluto de las ventas y mostrando cuánto se ha vendido junto a un beneficio.

## 2.3. Características del usuario

Existirán dos tipos de usuarios que puedan acceder a la aplicación: los usuarios normales (clientes) y el personal. Estos dos tendrán distintas funciones dentro de la aplicación, aunque tendrán varias en común. El personal espera con nuestra aplicación la gestión de la tienda, y el cliente lo que busca es una adquisición de productos más rápida. Este será el esquema de los tipos de usuarios que utilizarán nuestra aplicación.

## 2.4. Restricciones

### **Normativas, protocolos, estándares**

La aplicación debe de ser de escritorio (no web) y deberá incluir una GUI.

El lenguaje de implementación será Java.

La aplicación debe tener un diseño y una implementación sencillas para que cualquier persona que acuda a la tienda pueda usarla.

Fecha de entrega: 19/1/2017

- Especificación de requisitos
- Plan de proyecto

### **Limitaciones hardware**

No nos hace falta incorporar ningún hardware para poder utilizar nuestra tienda de discos.

### **Interfaces con otras aplicaciones**

La aplicación conectará a la BBDD automáticamente para la actualización de datos de la aplicación.

### **Funciones de control**

El control de la app se llevará a cabo en distintas máquinas que se encuentran en el establecimiento/tienda en el cual se tendrán que registrar siendo cliente o personal para poder utilizarlas.

### **Requisitos de lenguaje de programación**

La aplicación será desarrollada en Java.

### **Requisitos de fiabilidad**

El programa tendrá tolerancia a fallos, para que el sistema siga funcionando cuando aparezca uno.



### **Criticidad de la aplicación**

Indicará qué nivel de pérdida económica tenemos si la aplicación no está bien realizada. El nivel es formativo. Esto quiere decir que en realidad no perderemos dinero, pero si este proyecto si hubiese comercializado tendríamos que tener muy en cuenta que no podemos arriesgarnos a realizar una aplicación con defectos.

## **2.5. Supuestos y dependencias**

Esta aplicación no depende de otros sistemas externos lo que facilita su uso completo.

Es importante resaltar que en la maquina donde se instalara la aplicación debe disponer de sistemas operativos (como Windows y Linux).

Para la aplicación se intentará que sea accesible para el mayor número de personas posibles y, por esto, es necesario que se establezca en el mayor número de dispositivos, para que lo puedan usar tanto clientes habituales de Windows como de Linux.

El personal tendrá conocimientos y habilidades para poder controlar/modificar cualquier función básica de la aplicación en caso de emergencia.

## **2.6. Requisitos futuros**

Una de las ideas principales que se tienen sería dar soporte al market de SO para móvil como son Google Play y AppStore. De este modo, esta aplicación la podrían usar diferentes dueños de tiendas de discos y usarla en su establecimiento.

Respecto a las funcionalidades del sistema en sí, se podrían aumentar el número de funciones y conectarlas vía Internet. Un ejemplo sería la vinculación del sistema con distintas redes sociales para que el usuario pueda comparar los precios con otros establecimientos, buscar tienda más cercana...

# **3. Requisitos específicos**

## **3.1. Interfaces externos**

Las entradas que tiene el sistema son:

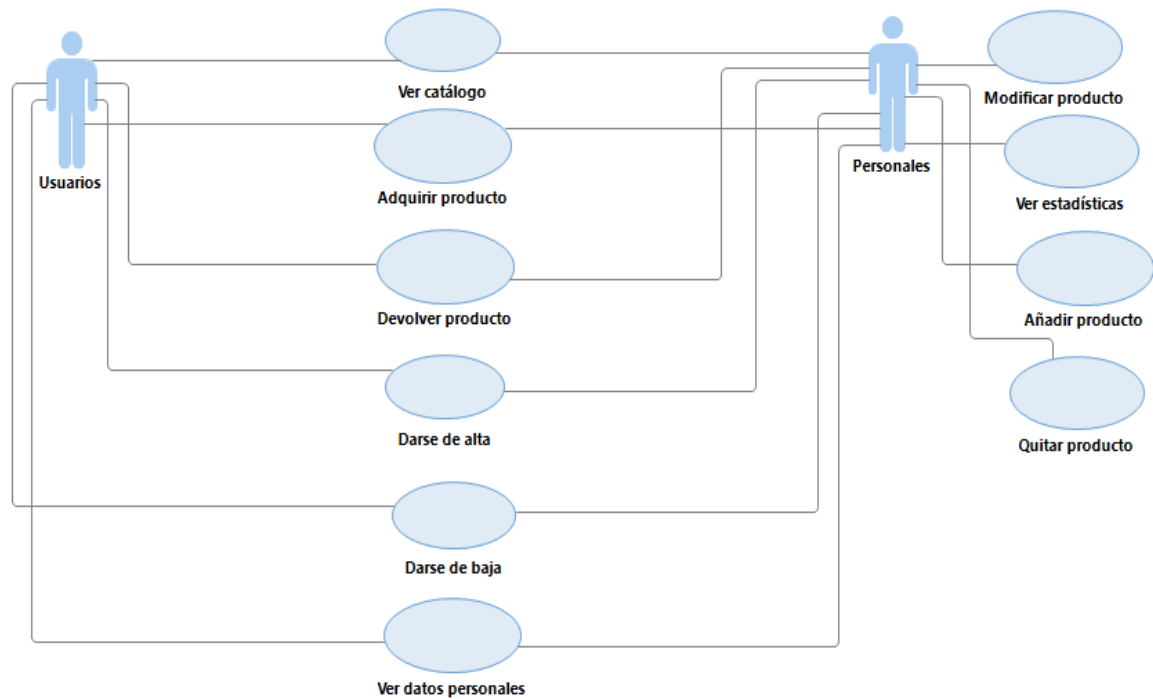
- **Un sistema de identificación de usuarios** para poder ofrecer las funcionalidades disponibles para cada tipo de usuario que haya en el sistema.
- **Menú con las funcionalidades principales del sistema.** Navegar por el catálogo musical, realizar compras, aplicar descuentos, etc. Las opciones variarán dependiendo de los privilegios que tenga la cuenta identificada en el sistema.
- Nuevo disco, pedido o usuario.
- Realizar compra.
- Modificar disco, pedido y usuario.

Y a continuación se listan las salidas:

- **Generar factura.** Mediante un ticket, o sin recibo, es decir, que se muestre en pantalla o también con la opción de ser enviada por correo electrónico.
- **Ver catálogo.** Vista de todos los discos disponibles.
- **Estadísticas.** Controlar los beneficios obtenidos y visualizar varias métricas de interés.
- **Búsquedas.** Resultados de búsquedas, a través del título de un disco, nombre del compositor, etc.

### 3.2. Requisitos funcionales

Casos de uso 'Usuarios' y 'Personal'



## Funciones del módulo de usuarios

Diagrama de actividad de la función ver catálogo

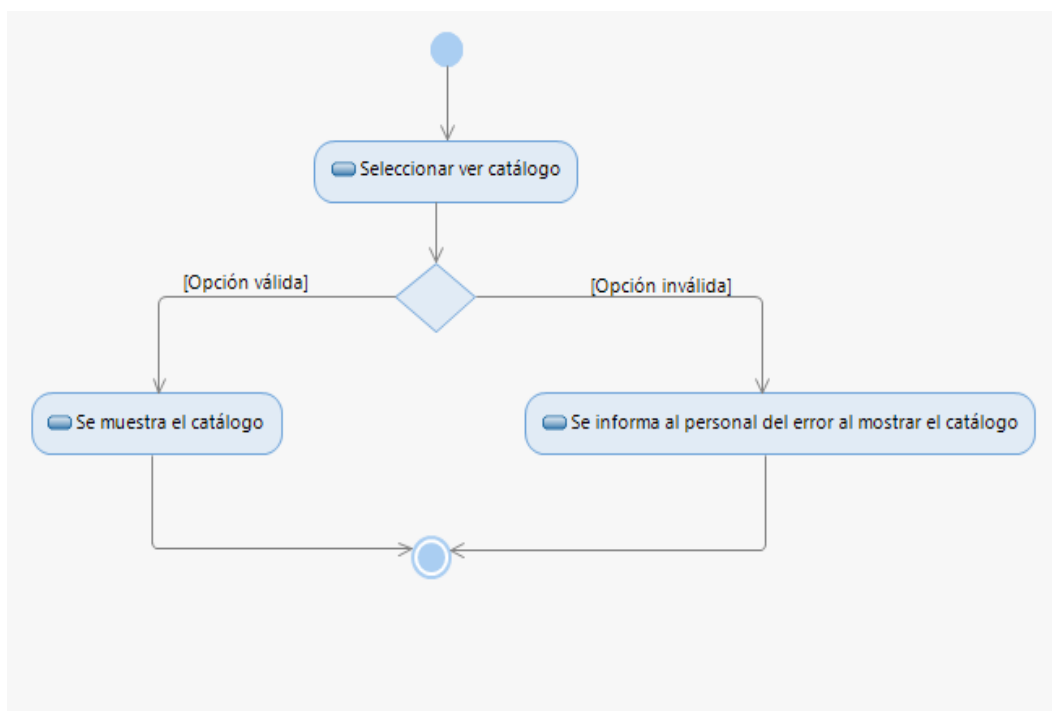


Diagrama de actividad de la función adquirir producto

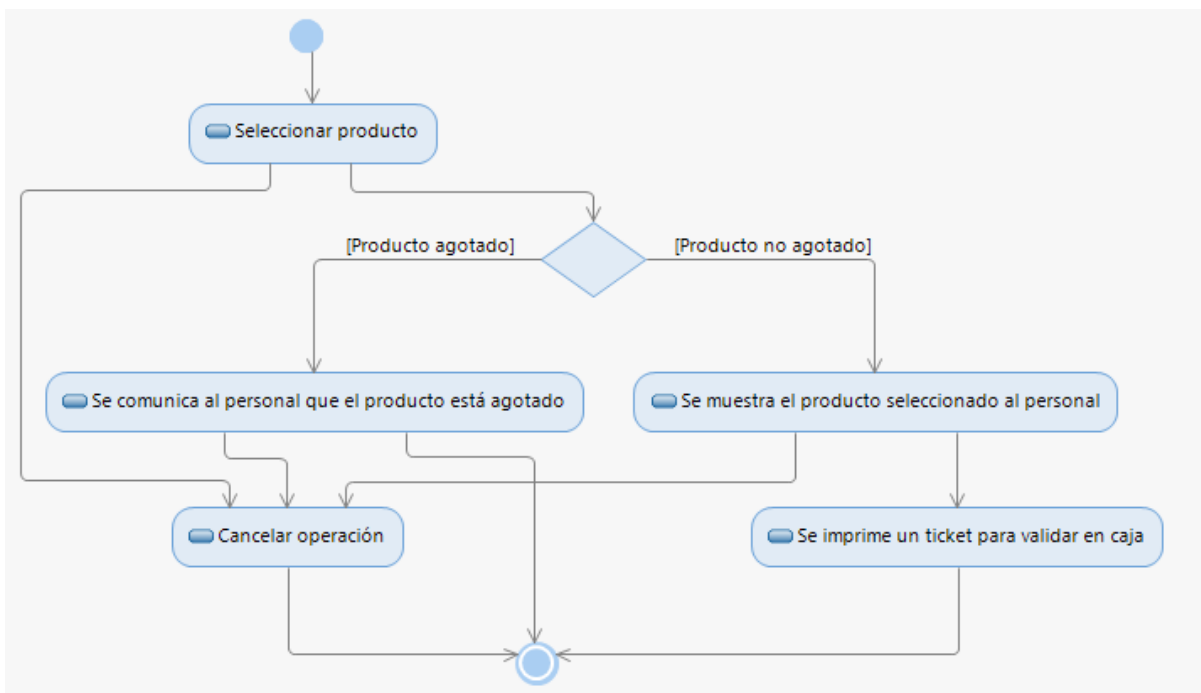


Diagrama de actividad de la función devolver producto

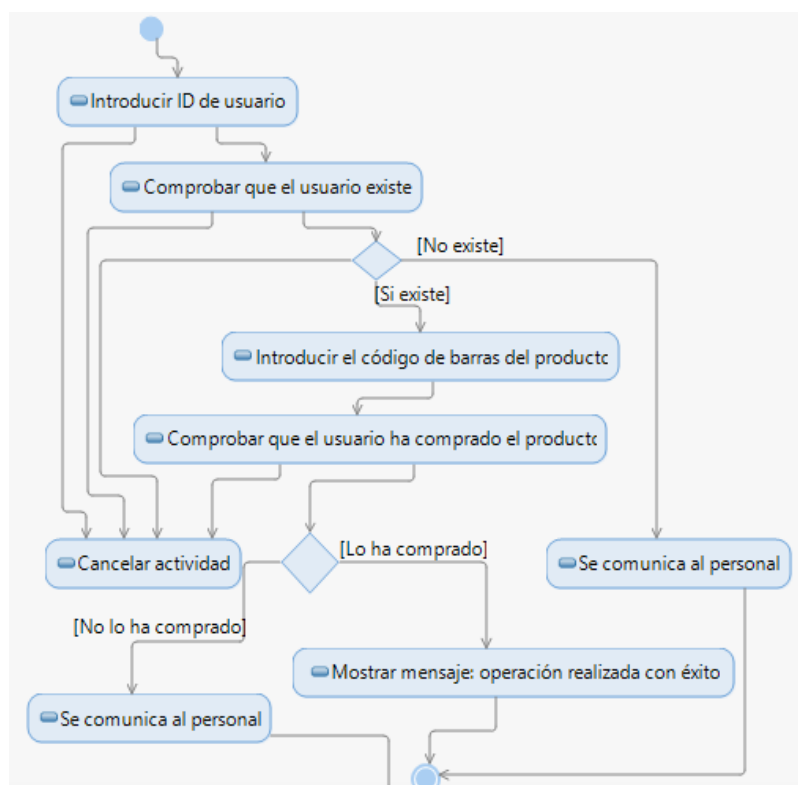


Diagrama de actividad de la función darse de alta

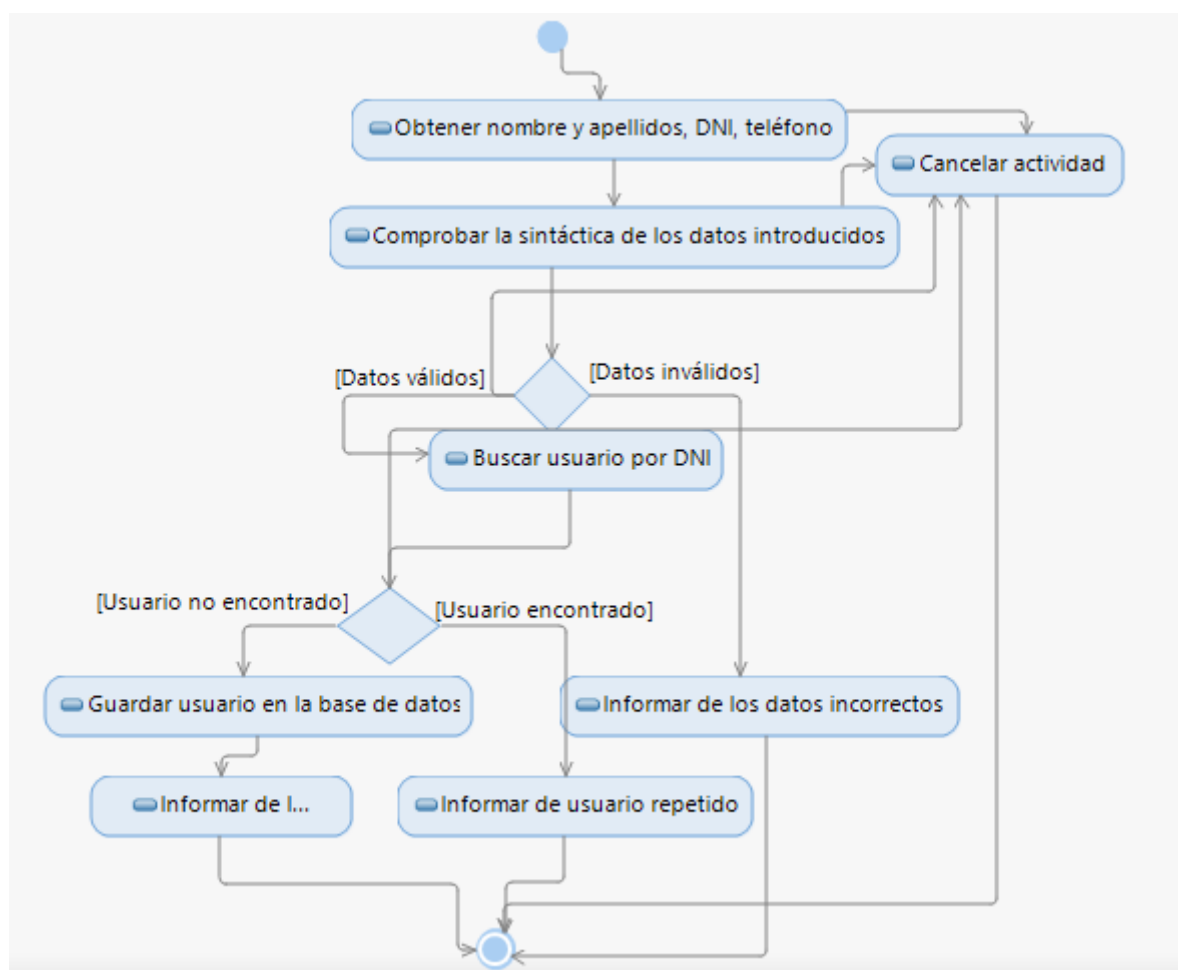


Diagrama de actividad de la función darse de baja

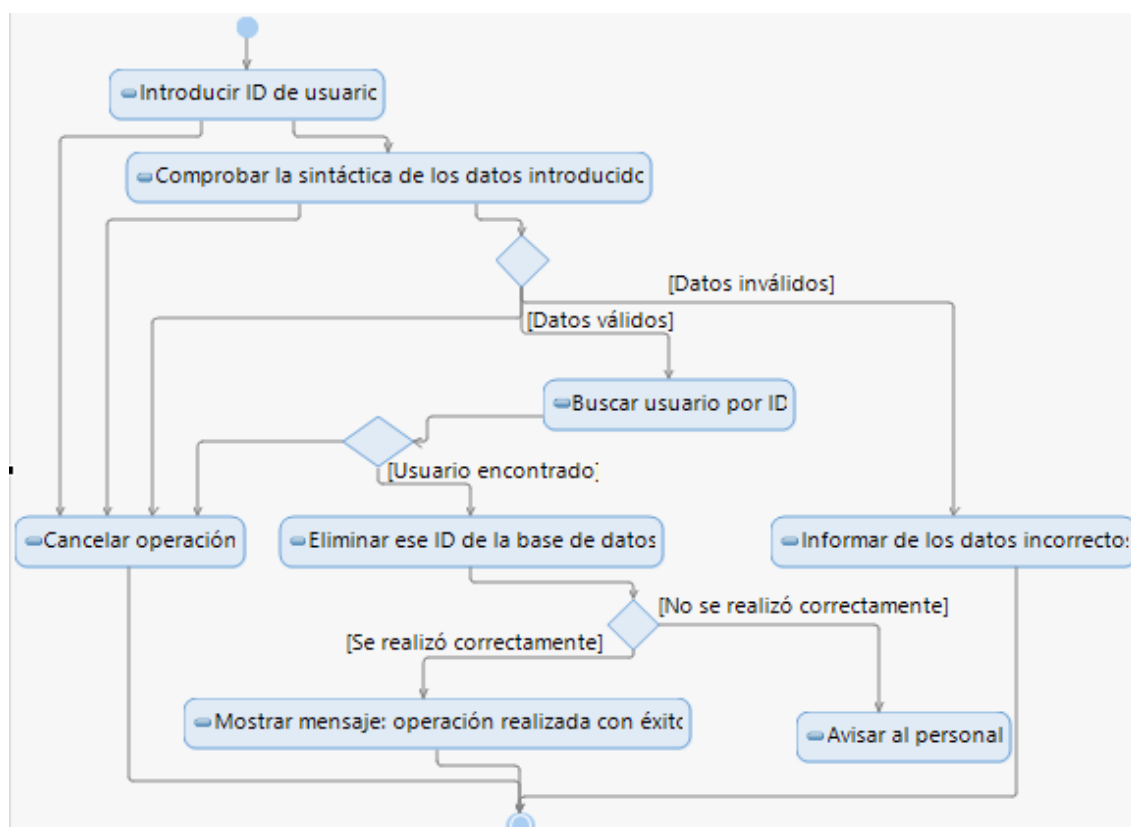
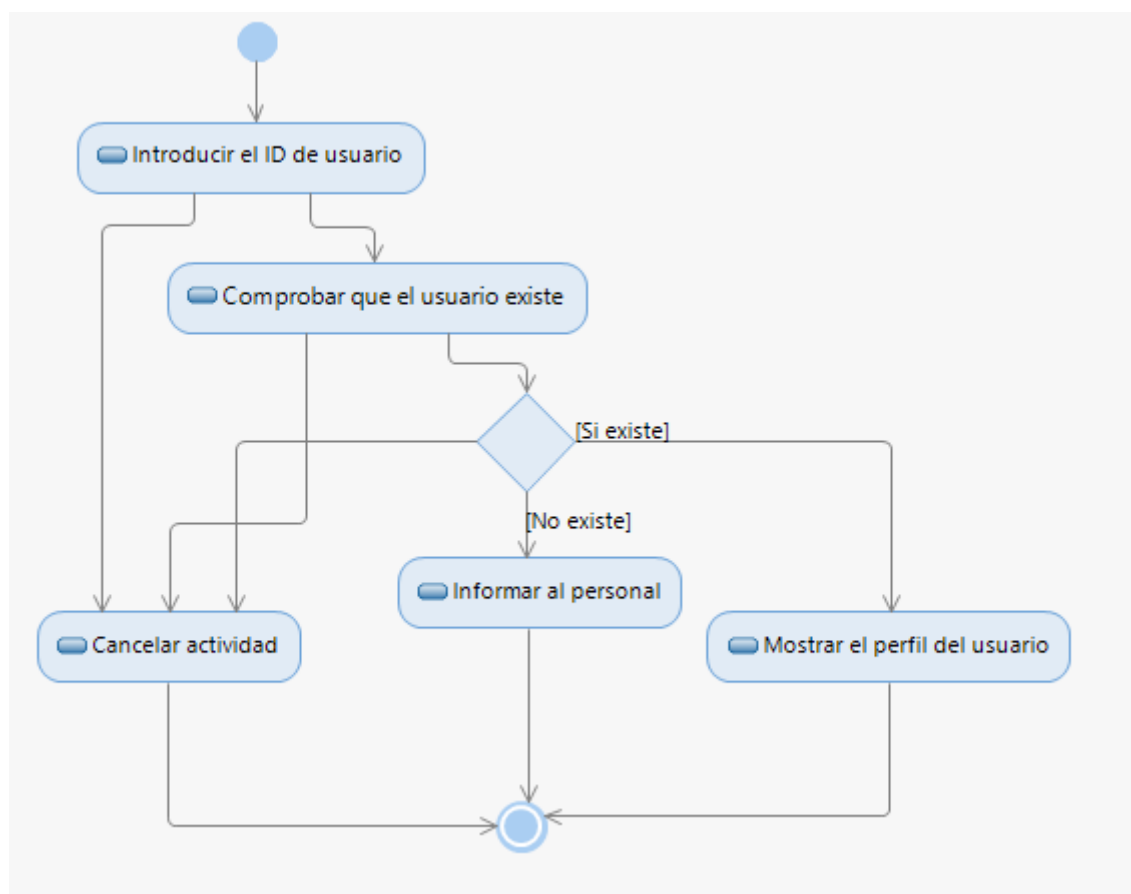


Diagrama de actividad de la función ver datos personales



## Funciones del módulo de personal

Diagrama de actividad de la función modificar producto

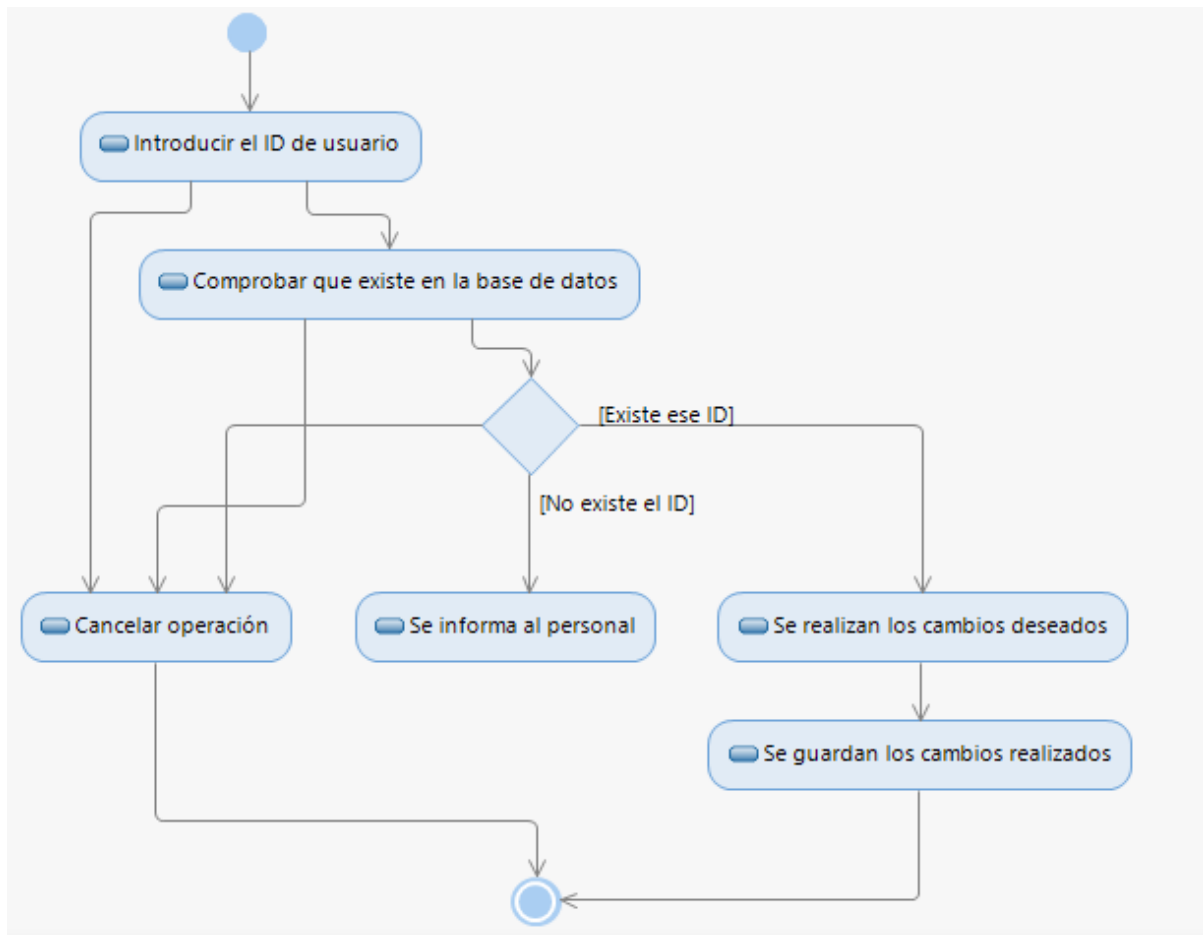




Diagrama de actividad de la función ver estadísticas

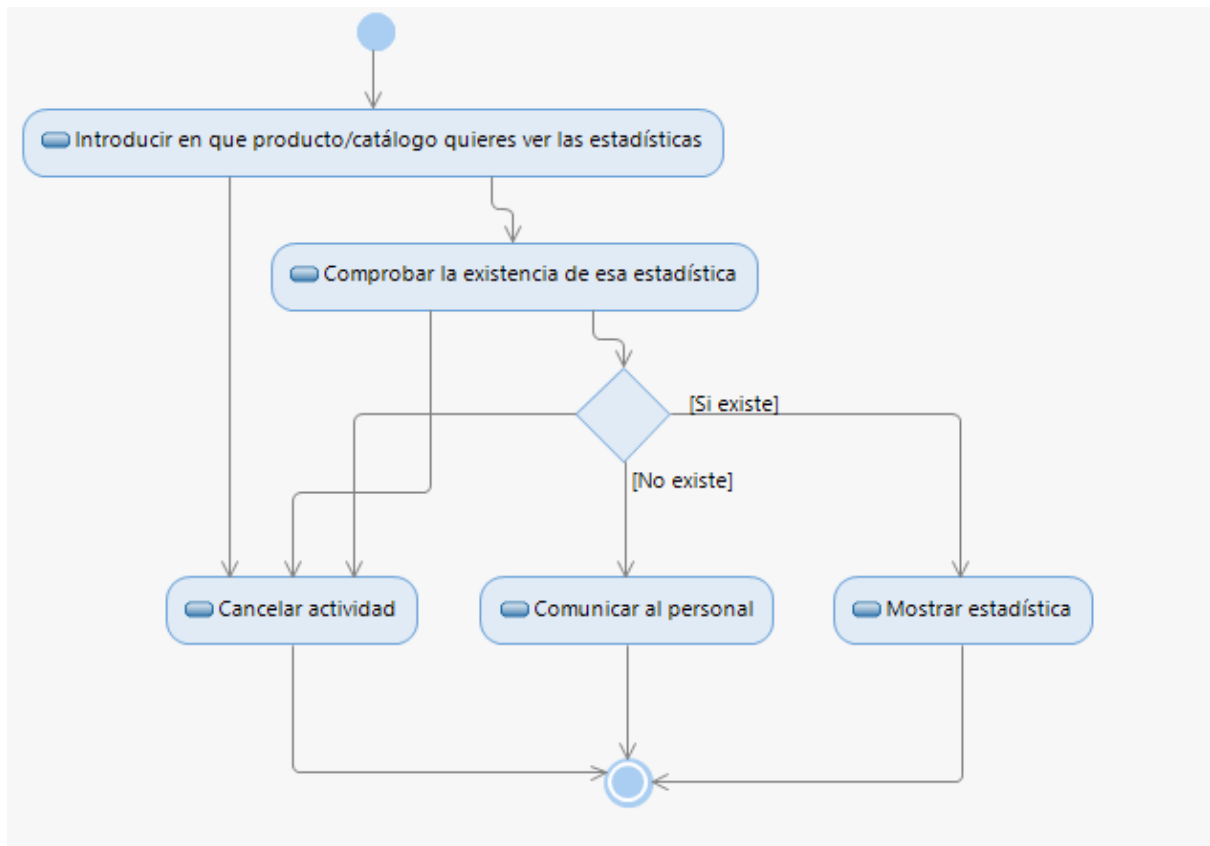


Diagrama de actividad de la función añadir producto

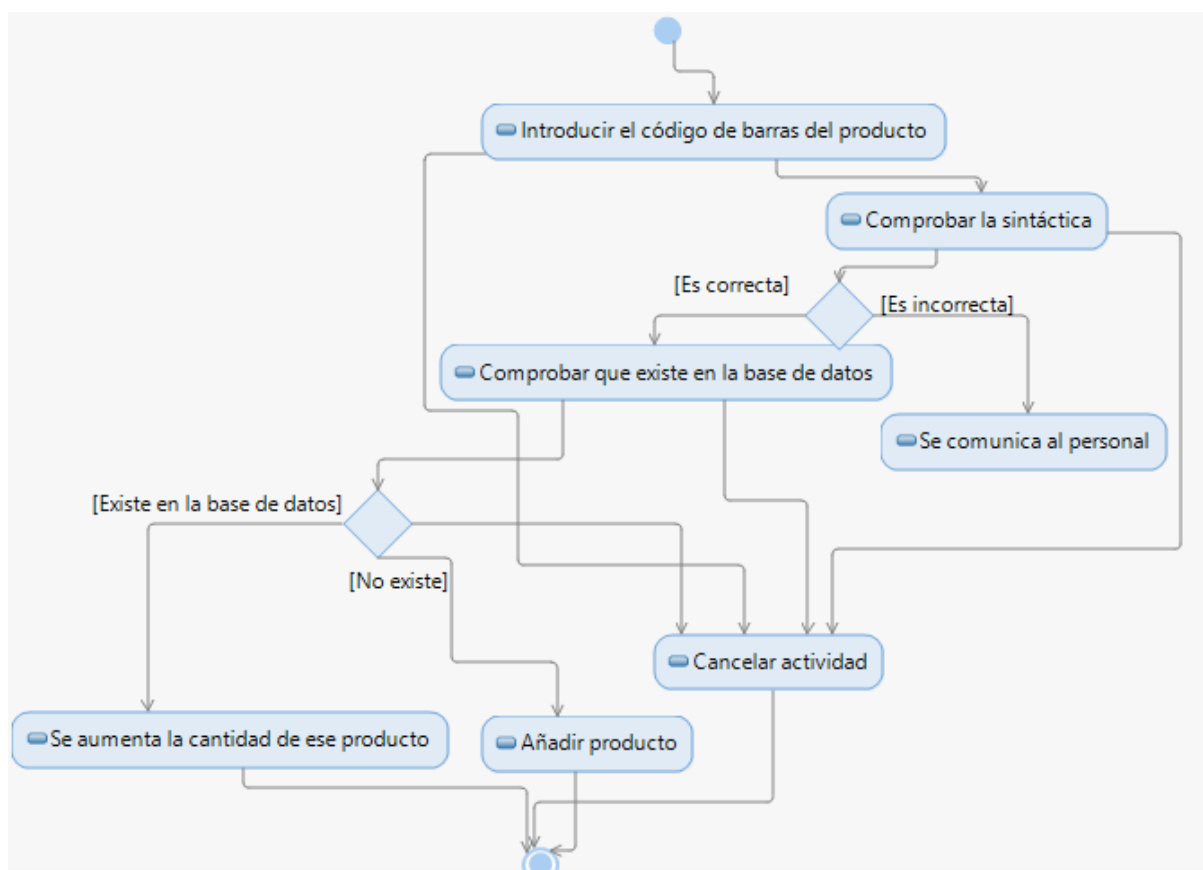


Diagrama de actividad de la función quitar producto

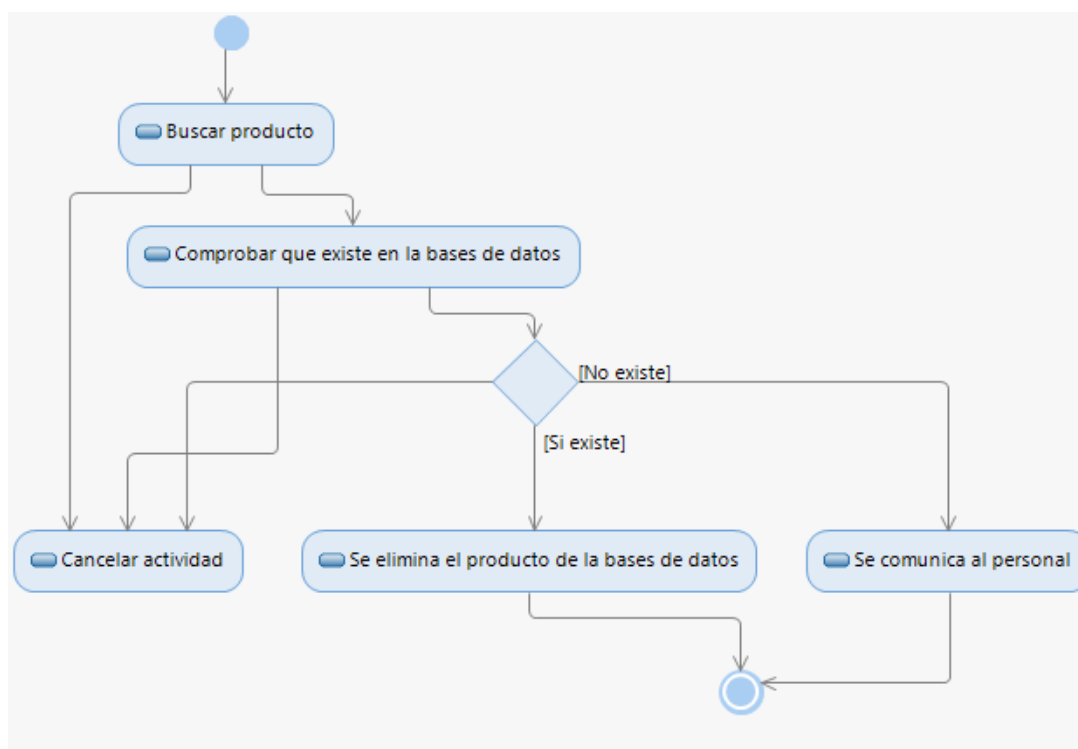


Diagrama de actividad de la función ver catálogo

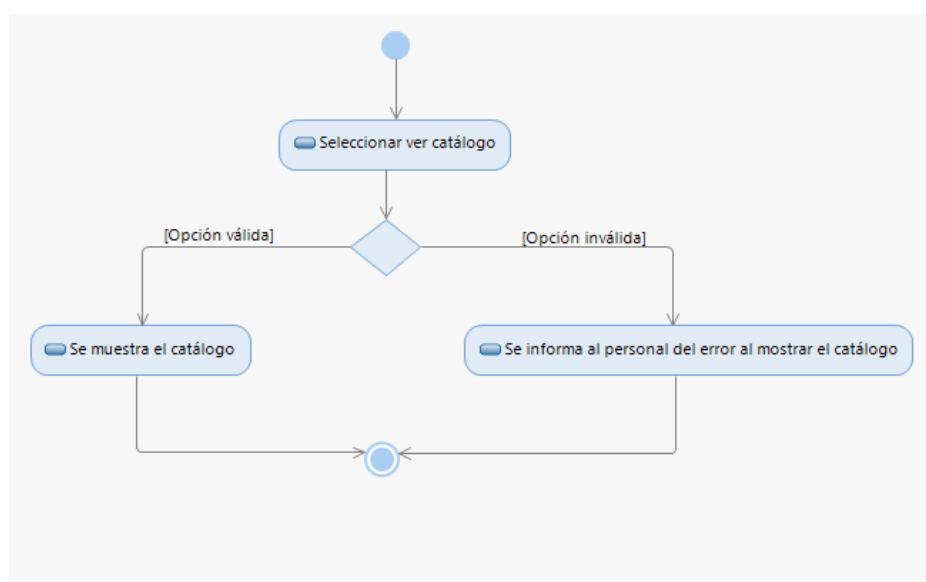


Diagrama de actividad de la función adquirir producto

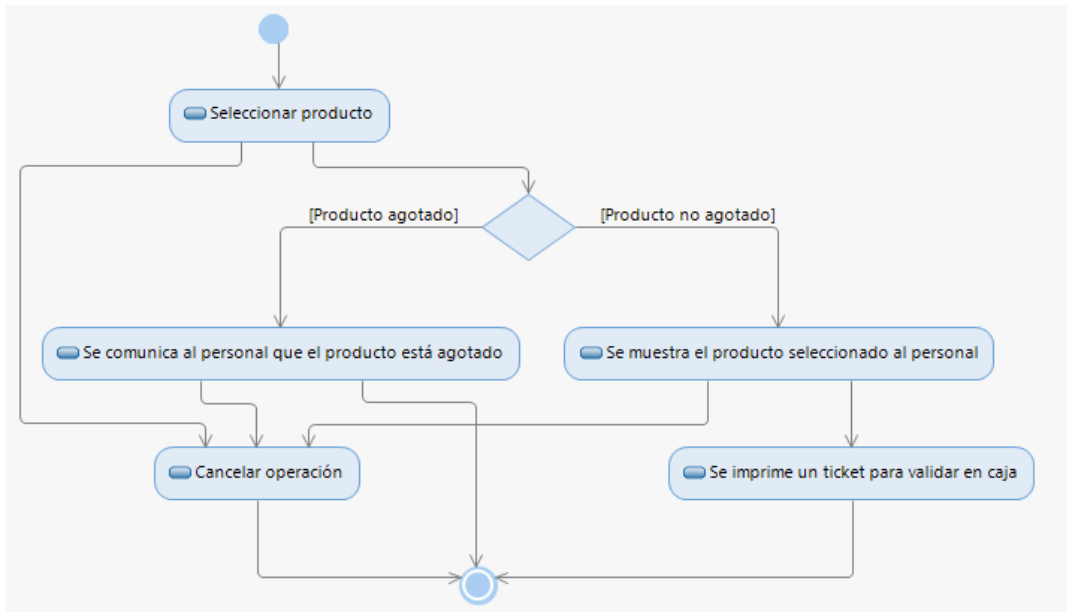


Diagrama de actividad de la función devolver producto

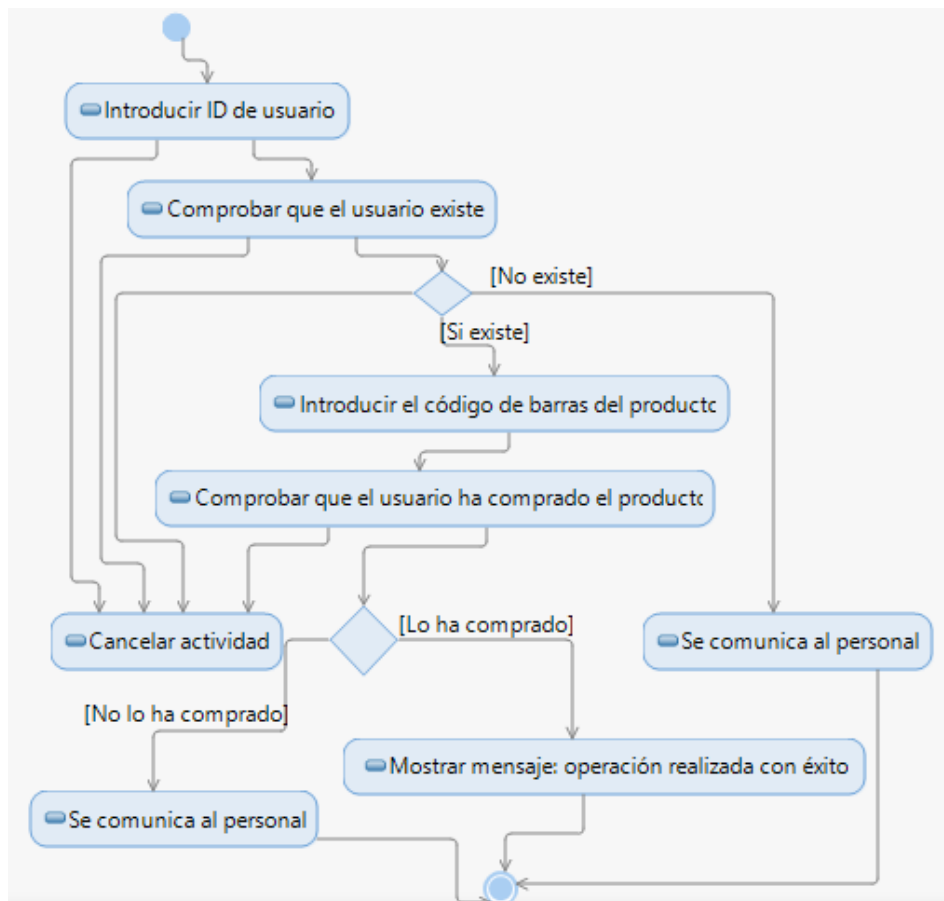


Diagrama de actividad de la función darse de alta

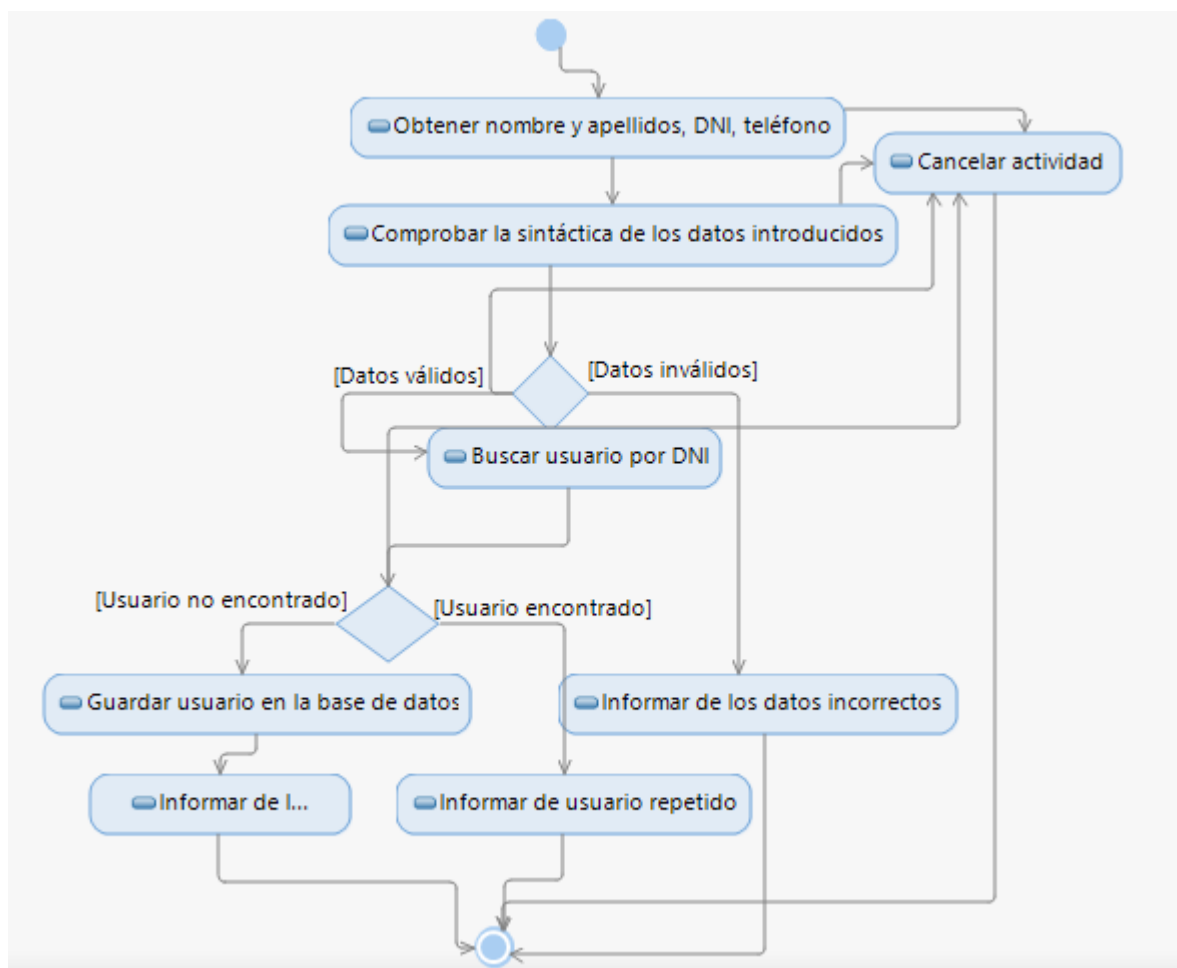


Diagrama de actividad de la función darse de baja

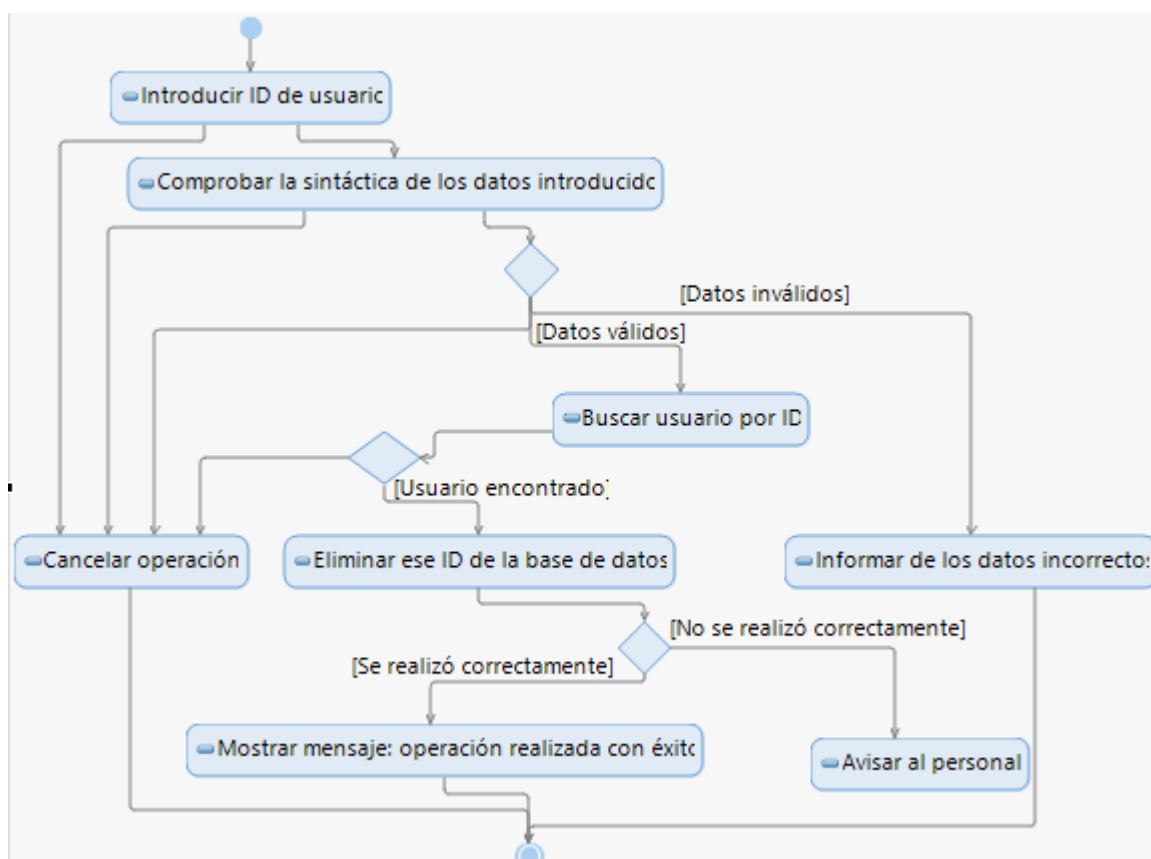
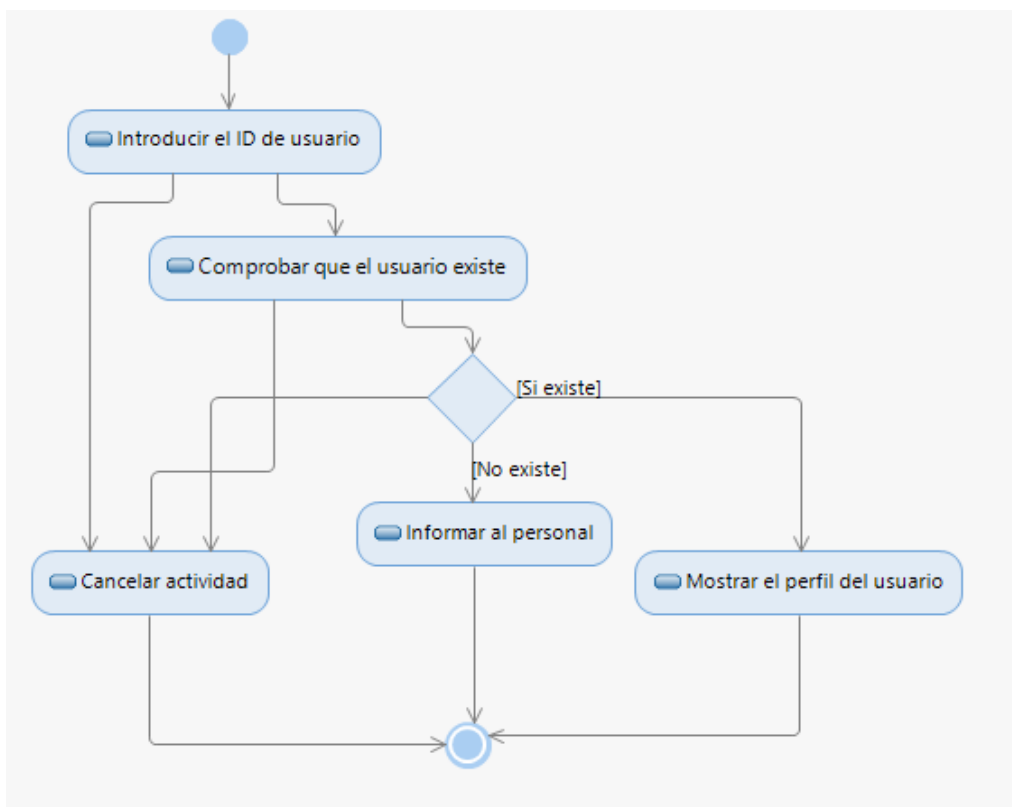
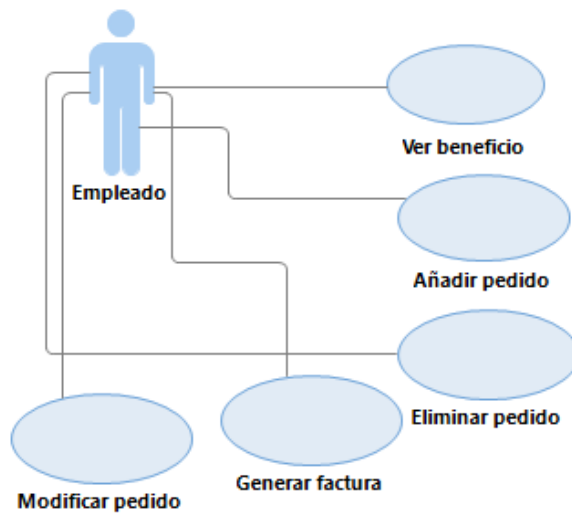


Diagrama de actividad de la función ver datos personales



## Casos de uso de 'Ventas'



## Funciones del módulo de ventas

Diagrama de actividad de la función ver beneficio

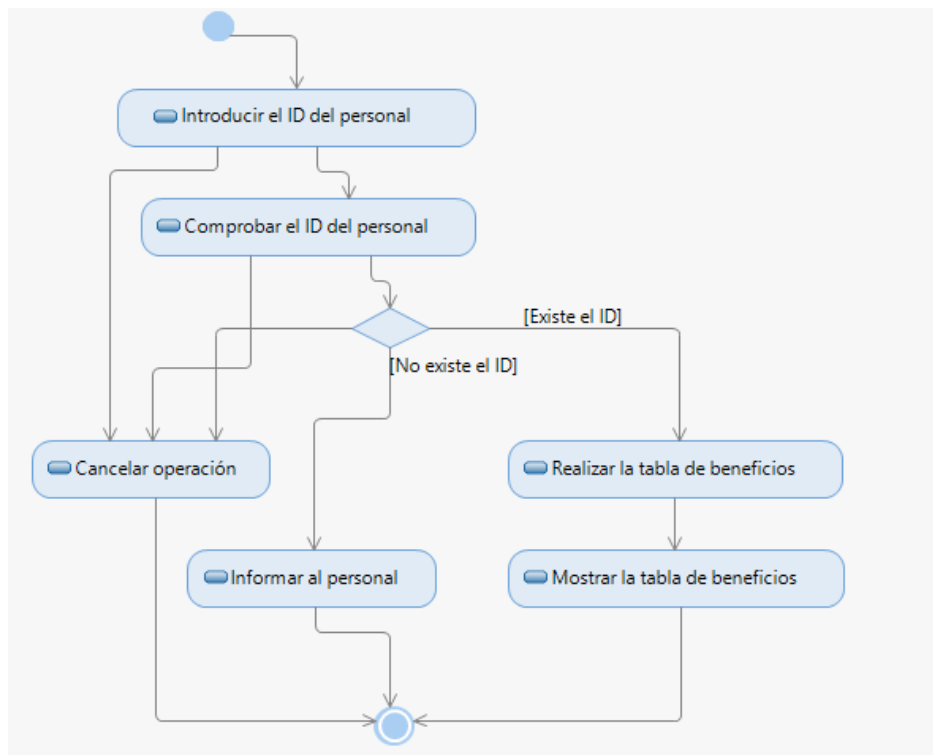




Diagrama de actividad de la función añadir pedido

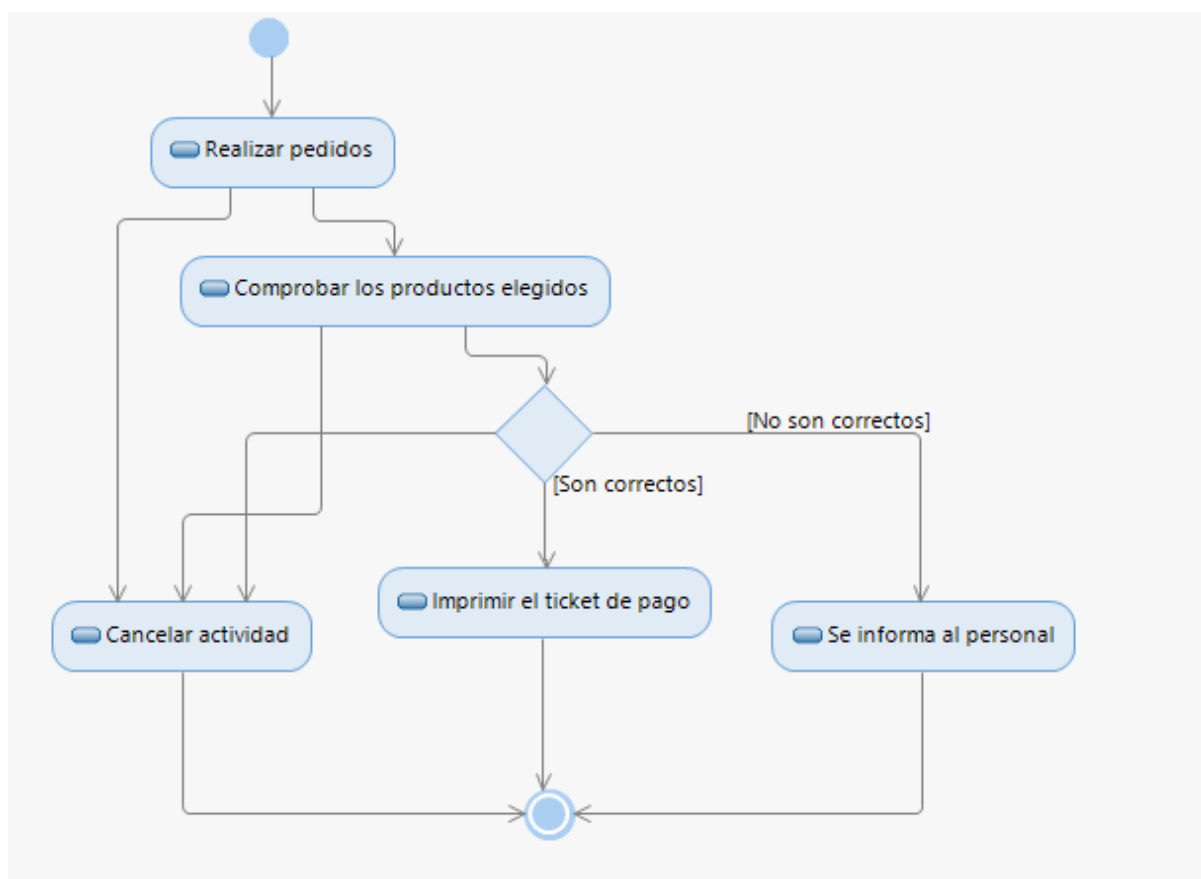


Diagrama de actividad de la función eliminar pedido

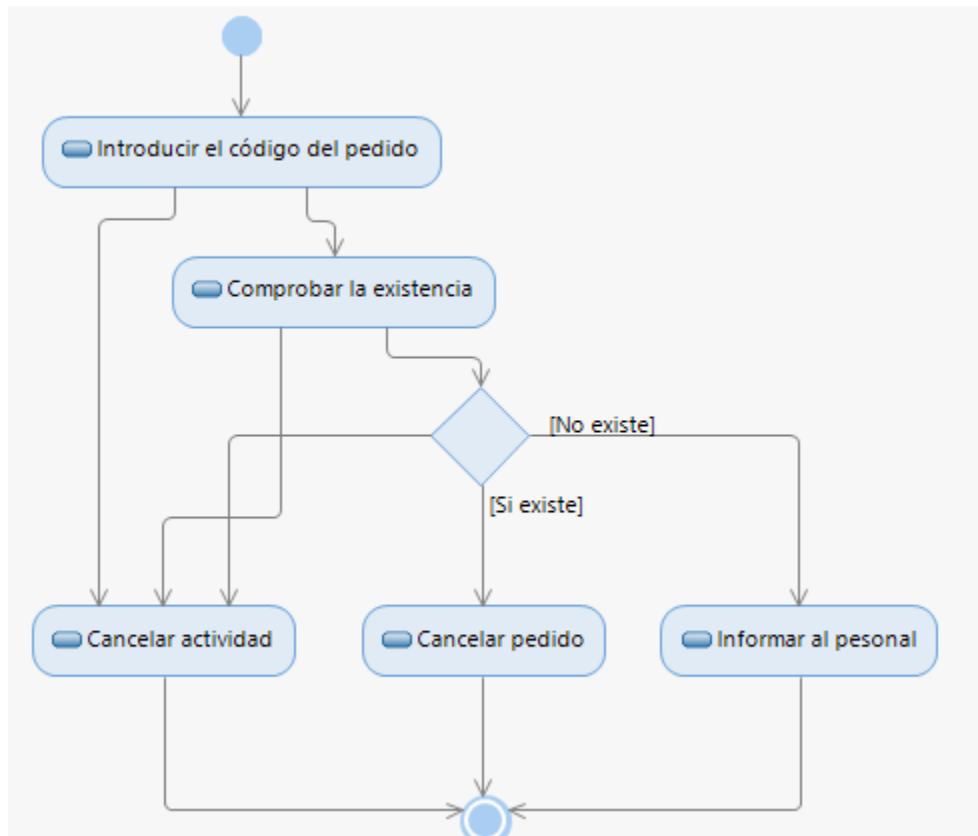


Diagrama de actividad de la función generar factura

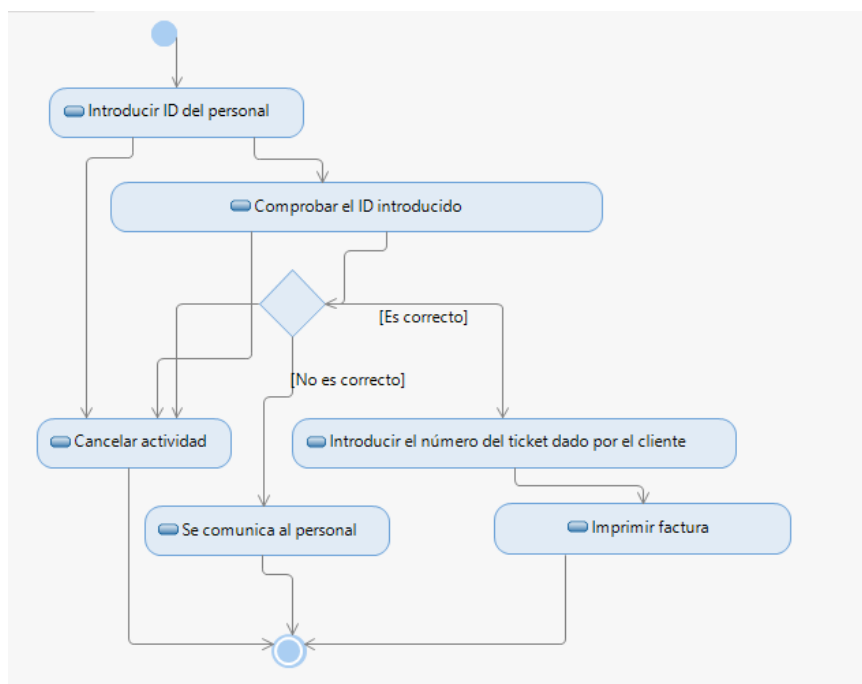
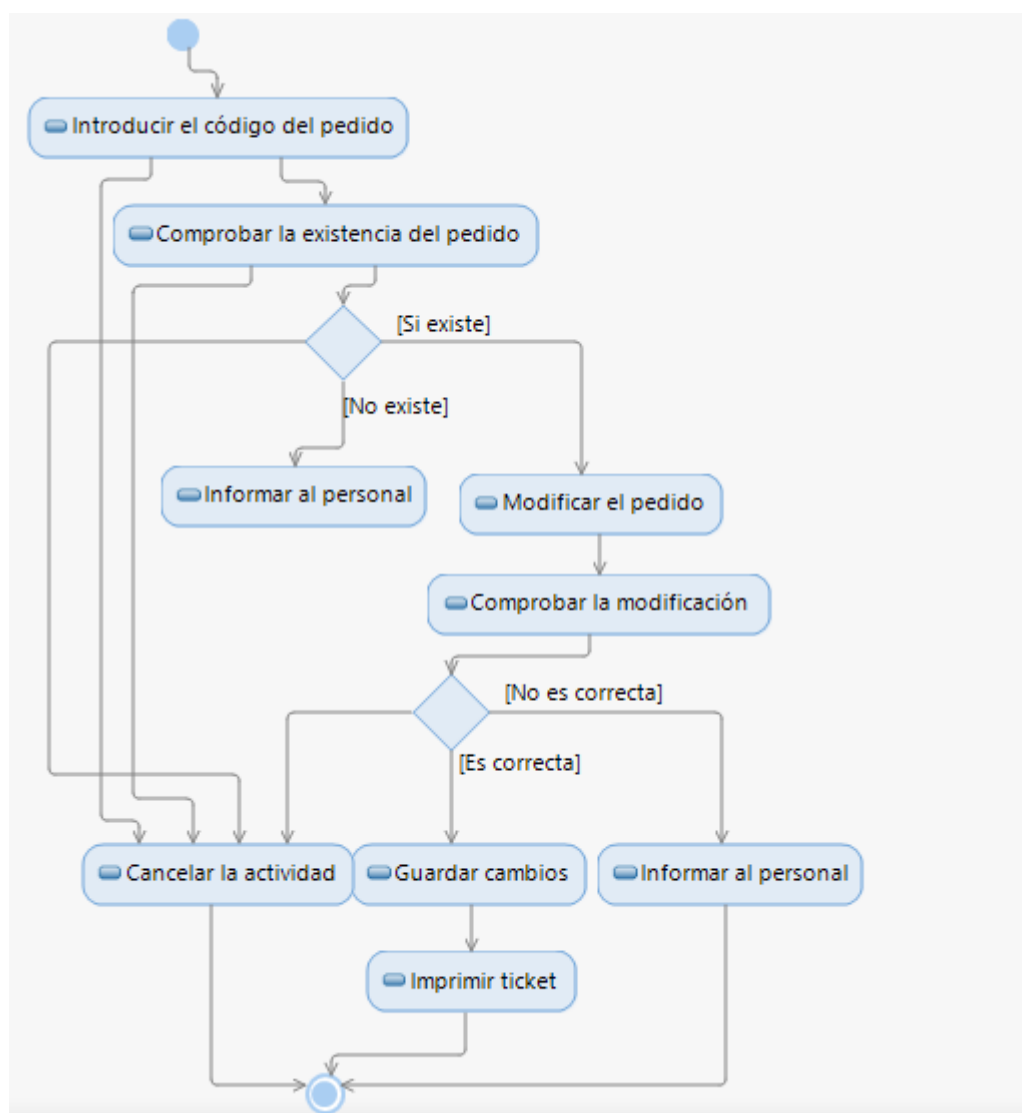
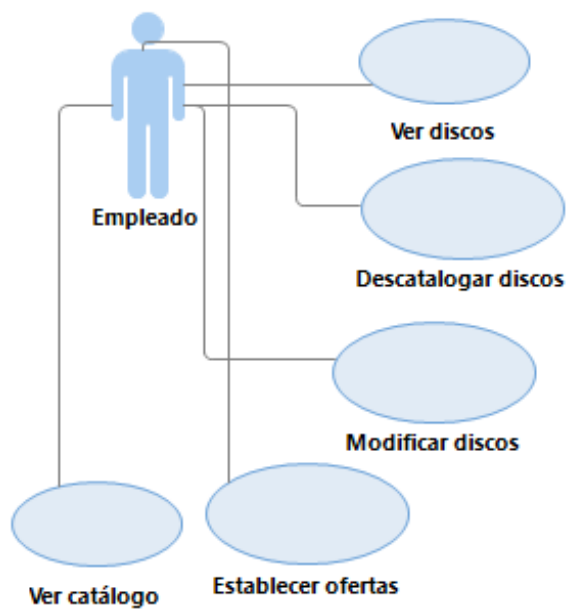


Diagrama de actividad de la función modificar pedido



## Casos de uso de 'Discos'



## Funciones del módulo de discos

Diagrama de actividad de la función ver disco

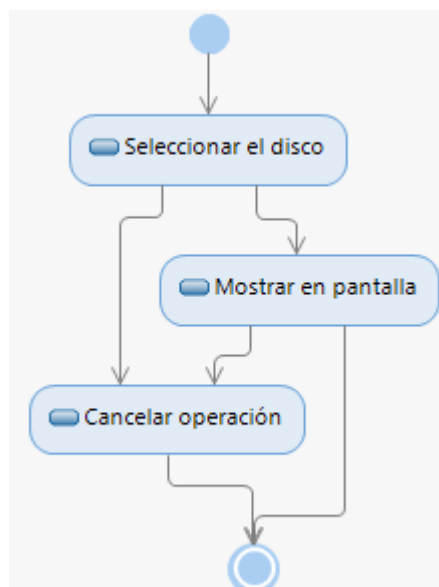


Diagrama de actividad de la función descatalogar discos

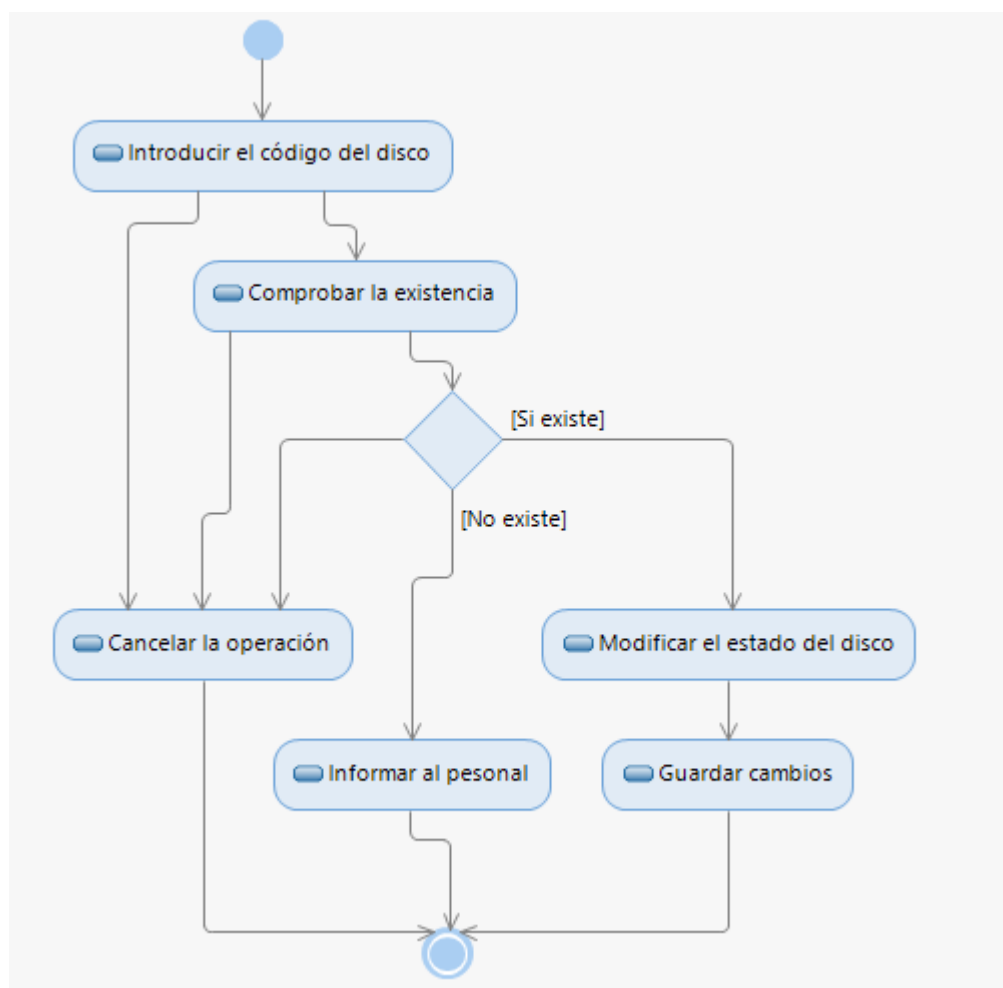


Diagrama de actividad de la función modificar disco

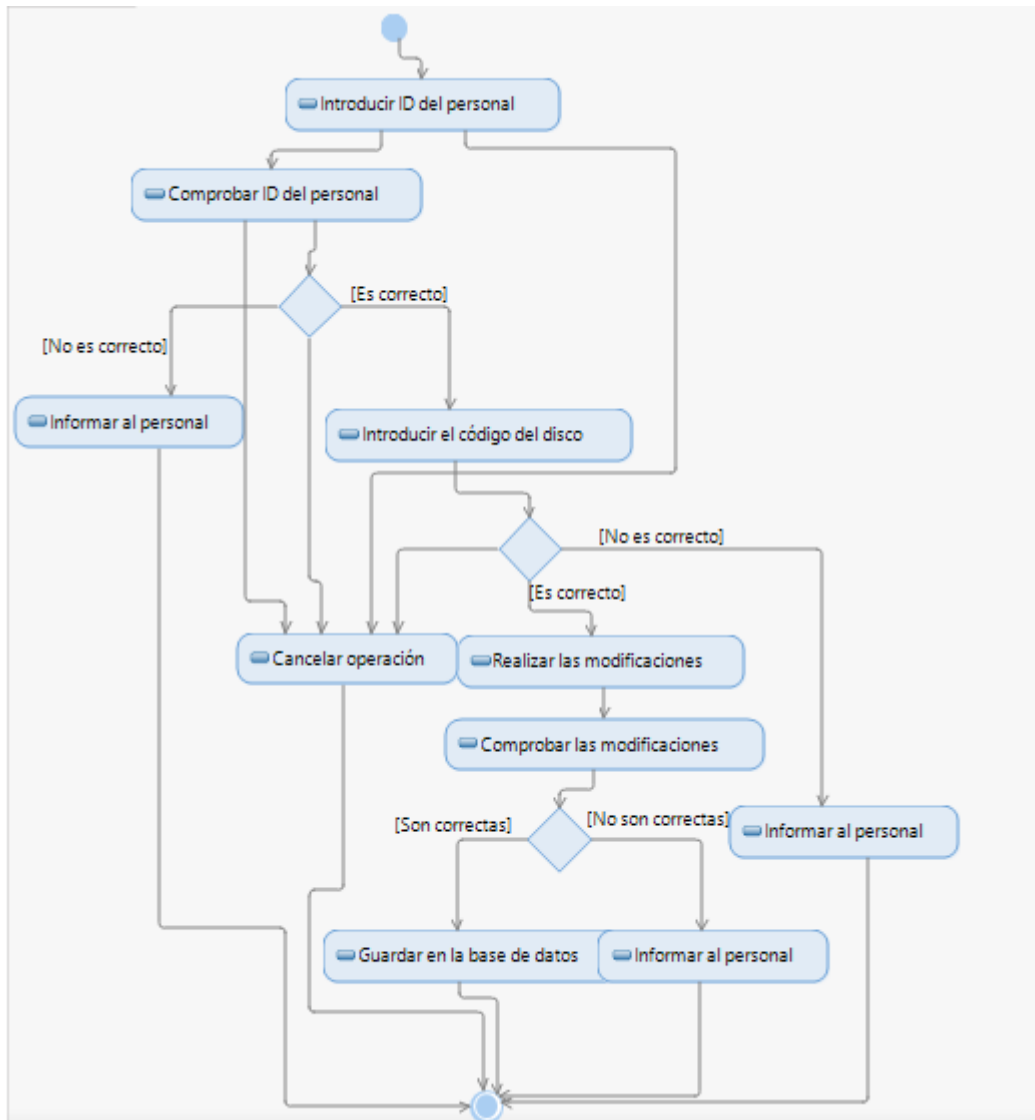


Diagrama de actividad de la función establecer ofertas

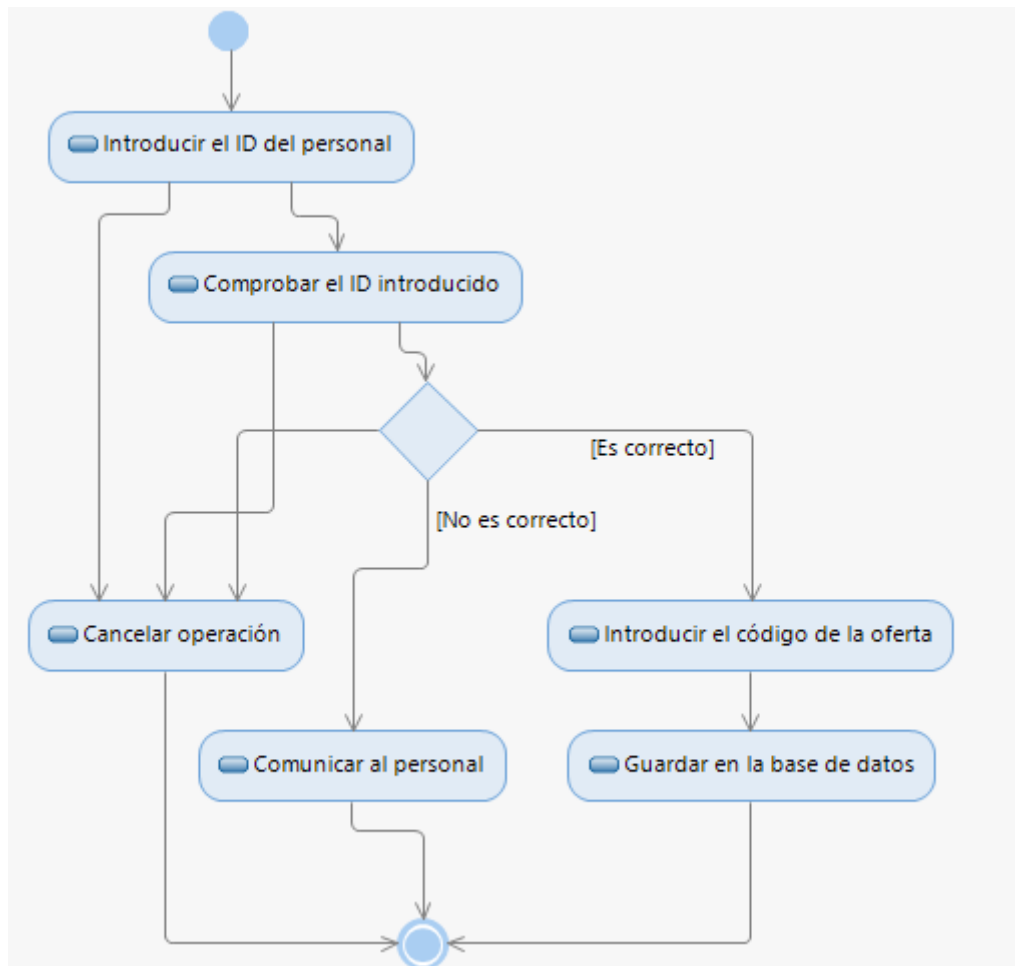
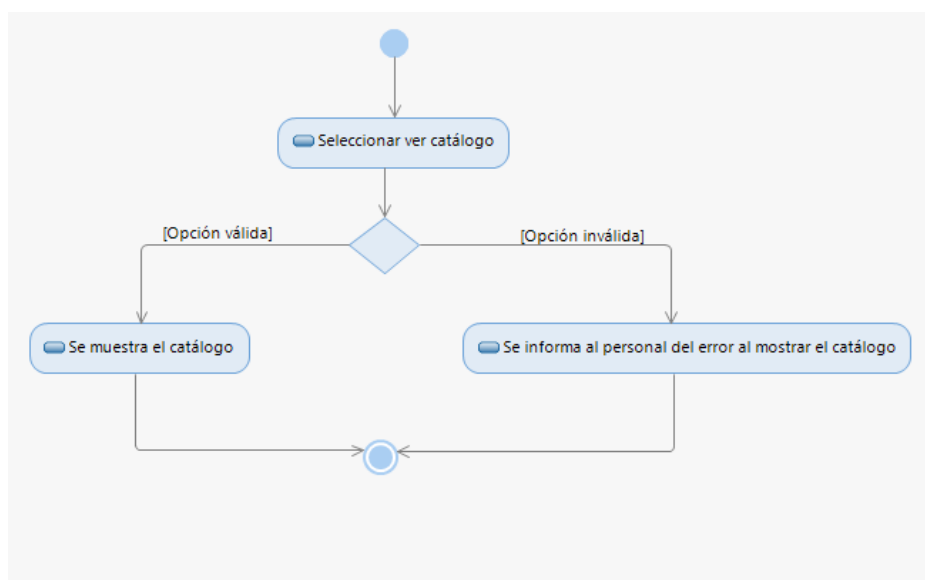


Diagrama de actividad de la función ver catálogo



### 3.3. Requisitos de rendimiento

I/O Records va a estar funcionando en varios terminales distribuidos por la tienda. El número de clientes o usuarios que se encuentran registrados va a ser de unos treinta para empezar.

Respecto al número de transacciones por segundo, veamos la siguiente tabla.

Tiempo (minutos)	Número de ventas
0 - 5	2
5 - 8	4
8 - 15	6
25 - 30	8

De esta manera podríamos seguir bajo esta suposición. El parámetro de transacciones por segundo tiene mucho más sentido para negocios que tienen muchas transacciones por segundo. En nuestro caso, nos fijamos en un espacio de 30 minutos (1800 segundos) y el número de transacciones efectuadas son 8.

### 3.4. Requisitos sobre la persistencia de datos

#### **Tipos de información utilizada por diversas funciones.**

Las diferentes funcionalidades de nuestro producto generan la necesidad de manejar distintos tipos de datos. Las funciones relacionadas con la administración de las cuentas de usuario van a hacer uso de datos como el nombre, la contraseña, el tipo de cuenta (si es de un cliente o si es de un empleado) y los datos personales del usuario (en caso de ser un cliente, se puede proporcionar la dirección del domicilio para recibir la compra). Este tipo de datos será clave para la correcta organización de la aplicación.

Asimismo, tenemos también un grupo de funcionalidades responsables de la gestión del inventario. Este tipo de funcionalidades manejan información relativa a los discos, como los nombres de los mismos, las canciones que incluye y su duración, el compositor y su precio de venta. Finalmente, las funciones relacionadas con las ventas y los beneficios, tendrán información sobre los pedidos e información contable sobre pérdidas y ganancias.

#### **Entidades de datos y sus relaciones**

De acuerdo con la definición de entidad de datos de la página web referenciada en el apartado Referencias, una entidad es una representación de un concepto de la vida real que se describe en una base de datos. Ese concepto de la vida real se describe mediante atributos. En nuestra aplicación existen diferentes entidades de datos que se pueden ordenar en tres tipos:



- La entidad Usuario, compuesta por toda la información de un usuario, ya sea un empleado o un cliente. Sus atributos serían el nombre, la contraseña, los datos personales y el tipo de usuario.
- La entidad Disco, que contiene todos los datos relacionados con el disco en cuestión. Así, el nombre del disco, el compositor, duración de las canciones y el precio de venta.
- La entidad Venta, formada por el conjunto de características de una venta. Sus atributos son el identificador de venta, discos pedidos, con sus respectivos precios, cantidades solicitadas y un atributo que indique si la recogida es en tienda o a domicilio.

Estas tres entidades son los pilares de datos que existen en nuestra aplicación. Bajo el punto de vista de un cliente, la entidad Usuario se relacionaría con Disco en el momento en el cual el cliente consulte el catálogo de discos para realizar una compra. El cliente elige el disco deseado, se efectúa la venta y se actualiza el inventario.

Bajo el punto de vista de un empleado, la entidad Usuario se relacionaría con Disco cuando se quisiese modificar o verificar el inventario, los precios y las ofertas de los discos. A su vez, la entidad Usuario se relacionaría con Pedido en el momento que el empleado modificara o comprobase los pedidos de la tienda.

#### **Frecuencia de uso**

Los datos de nuestro producto serán frecuentemente consultados debido a que el empleado los utilizará para gestionar su negocio, y al mismo tiempo, los clientes para hacer compras. Esto quiere decir, que, durante el horario de apertura, los datos estarán en constante uso. Tanto los datos de usuario, los de los discos y los de las ventas.

#### **Formas de acceso**

Nuestro producto solo permite el acceso a los datos mediante el uso de una cuenta de usuario. Sin embargo, un cliente podrá visualizar la aplicación en calidad de invitado sin necesidad de usar la aplicación.

Asimismo, no todos los empleados podrán acceder a los datos de ganancias y pérdidas de la empresa.

#### **Restricciones de integridad**

Ya que nuestro producto gira en torno a una base de datos, es importante en la medida de lo posible preservar los datos de la misma.

Cuando se diseña una base de datos se debe reflejar fielmente el universo de discurso que estamos tratando, o lo que es lo mismo, reflejar las restricciones existentes en el mundo real. Para ello vamos a definir una serie de restricciones de integridad de los datos.

- La operación de actualización (inserción, borrado o eliminación) cuya ejecución ha de dar lugar a la comprobación del cumplimiento de la restricción.

- La condición que debe cumplirse, la cual es en general una proposición lógica. Puede tomar como valor: verdadero o falso.
- La acción que debe llevarse a cabo dependiendo del resultado de la condición.

Para la inserción, borrado o eliminación de un usuario, no existen muchas restricciones. En caso de insertar un usuario, sus atributos serán el nombre de usuario, la contraseña, los datos personales y el tipo de usuario. Con lo cual la condición que se debe cumplir es que tanto el nombre, la contraseña y los datos personales no existan ya en la base de datos. Si no existen se inserta ese nuevo usuario y si existen se muestra un mensaje de error. Para el borrado y la eliminación no existen restricciones.

Si queremos insertar un nuevo disco, sus atributos son el nombre del mismo, el precio, el número de existencias, y datos de interés como la lista de canciones, el compositor, discográfica, etc. Una de las condiciones es que no exista ya ese disco en el catálogo. Si ya existe, se muestra un mensaje de error. En cuanto a la cantidad de existencias, sólo se podrá insertar si no se supera el límite de capacidad del almacén. Si se intenta insertar algún producto más, se muestra el correspondiente mensaje de error. Los demás atributos no tienen restricciones de inserción. Para el borrado y la eliminación de discos, la restricción existente es que sólo un usuario con los privilegios suficientes puede modificar o eliminar discos.

Para la inserción, borrado o eliminación de un pedido, las restricciones son las siguientes. Si hacemos un pedido superior a la cantidad disponible en stock, se muestra un mensaje de error. Para solicitar que el envío se haga a domicilio, se debe haber introducido antes en la cuenta la dirección. Si esto no ha ocurrido, se muestra un mensaje de error. Para borrar o eliminar un pedido no existe ninguna restricción.

#### **Requisitos de retención de datos**

Si un usuario lleva más de 2 años sin utilizar su cuenta, esta se eliminará automáticamente.

### **3.5. Restricciones de diseño**

A continuación, se especifica el hardware y las restricciones del sistema que va a ejecutar la aplicación.

**Memoria:** En cuanto a memoria, nuestros dispositivos constan de una memoria interna de 250GB. Creemos que es un tamaño adecuado, ya que en ella se guardará la aplicación, con su base de datos de productos y precios correspondientes. También se guardarán las cuentas de usuario, tanto de los clientes como las de los empleados. En caso de necesitar ampliar la memoria, se podría introducir un disco duro para aumentarla, pero creemos que con el tamaño propuesto no debería haber problemas.

**Sonido:** El dispositivo no va a tener tarjeta de sonido. Es algo innecesario que no repercute en la funcionalidad del producto. Por tanto, no hemos incluido audio en nuestra aplicación.

**Usuarios:** La idea es que no exista límite de usuarios. Al no existir un límite, las cuentas a

las que no se haya accedido en un periodo de 2 años serán eliminadas. De esta forma hacemos que se optimice el espacio disponible, y que se tenga únicamente ocupado el necesario.

**Red:** Disponemos de una conexión a internet de fibra óptica con un ancho de banda de 100MB. Nuestros dispositivos estarán conectados vía Ethernet con cables URJ45. Si el proveedor perdiera la conexión en la zona o en su terminal, tendríamos un problema, ya que carecemos de alternativas para poder conectarnos a Internet. Lo mismo ocurre si falla nuestro cable URJ45, pero en ese caso sí tendríamos repuestos en la tienda para poder reemplazarlo.

**Características de los equipos:** Nuestras computadoras están equipadas con un procesador Intel Core i5 a 2.7 GHz, con 4GB de memoria RAM y no tienen una tarjeta gráfica dedicada, ya que, no es necesaria para ejecutar con la suficiente fluidez nuestra aplicación.

Son ordenadores con características suficientemente potentes como para cumplir con los requisitos del programa. En caso de querer ampliar o mejorar el rendimiento de la aplicación, deberíamos cambiar la máquina, pero para nuestro diseño principal, con estas especificaciones es suficiente.

### 3.6. Atributos del sistema software

**Fiabilidad:** Nuestra idea es que nuestro producto tenga una fiabilidad de al menos 5 años sin necesidad de cambios.

En un principio, consta de un software sencillo, que se adapta a cualquier máquina, pero esto depende de varios factores. Si la temperatura de nuestro dispositivo es mayor de 80°C, esto puede propiciar que el software funcione de una forma más lenta de lo normal. Otro factor es el tiempo que el software está operativo. Es aconsejable que se apague después de cada jornada laboral para hacer que dure más tiempo.

**Disponibilidad:** Es un concepto que se utiliza para ver si un sistema es capaz de llevar a cabo sus funciones, teniendo en cuenta sus horas de uso. Para ello vamos a tener un porcentaje de disponibilidad del 99%, es decir, si la máquina está todos los días del año en funcionamiento, tendríamos un tiempo de inactividad de 16,76 horas.

**Seguridad:** Para la seguridad de la aplicación, tenemos que tener en cuenta que nuestros datos, tales como las distintas cuentas de los usuarios, se pueden perder si sucediera cualquier tipo de catástrofe. Por ello, realizamos copias de seguridad mensualmente. Para evitar que se creen cuentas de administrador ajenas a los empleados, es necesario introducir una clave maestra previamente, que solo el jefe del local conocerá.

**Mantenibilidad:** El software tiene la facilidad de agregar funciones nuevas en caso que

fuera necesario, así como una forma eficiente y rápida de corrección de errores y una rápida adaptación a futuros sistemas operativos. Se puede modificar de manera sencilla para solventar posibles fallos de funcionalidad.

**Portabilidad:** Nuestra aplicación es multiplataforma, se puede utilizar en cualquier sistema, tanto de Windows como de Linux o Mac OS. Es reutilizable, es decir, se puede modificar sin tener la necesidad de crear un programa completamente nuevo.

## 4. Apéndices

Un plan de futuro que hemos planteado en el equipo es el de desarrollar una tienda online. Sin embargo, nos parecía una idea un poco ambiciosa debido a las limitaciones de tiempo y a la poca experiencia en este terreno. Investigando acerca del tema, hemos dado con distintas plataformas para comercio electrónico como *Magento*, *PrestaShop*, *WooCommerce* o *Shopify*. Utilizando alguna de estas herramientas el proceso de desarrollo sería mucho más rápido ya que no habría que empezar desde cero. Esta idea la queremos dejar reflejada en el documento por si en un futuro se quisiera llevar a cabo.