

**Grado en Ingeniería de Computadores – Curso 2020-21**

**Ingeniería del Software: Laboratorio**

**Cuaderno de prácticas. P2 - Análisis. (Modelo conceptual de clases)**

**Solución del ejercicio práctico:**

*Primero se ha hablado con el responsable de la tienda, quien nos manifiesta que cuando se atienda a un cliente en la tienda porque quiere pedir libros, debe poder generar un pedido. Se comprobarán las existencias de dicho libro, en caso de que todos los libros del pedido se hallen disponibles, se realizará el pedido, en el que se introducirá un código de pedido, la fecha se tomará del sistema, el DNI del cliente y por cada libro pedido, su código y cantidad de ejemplares. Una vez que el cliente paga, el sistema debe generar la factura de venta correspondiente al pedido, que incluye un número de factura, el código del pedido al que corresponde, el DNI, la fecha de emisión, el nombre y domicilio del cliente, y la lista de todos los libros facturados (por cada libro, su código, título, cantidad de ejemplares, y precio de cada ejemplar), y el precio final de la venta. Si no existe algún libro, se generará una orden de impresión a la imprenta, posponiéndose la entrega del pedido hasta que todos los títulos solicitados se hallen en stock. En cualquier momento se podrán recuperar los datos de un pedido pendiente y se procederá desde ese momento de la misma forma como en el caso de hacer un nuevo pedido, utilizando las mismas pantallas (comprobar existencias, orden de impresión, facturación). Los pedidos pendientes serán susceptibles de ser cancelados en cualquier momento.*

*En la tienda hay un empleado que usará también la aplicación, a quien nos remite el responsable para más detalles. El empleado nos aclara que una orden de impresión a la imprenta debe incluir un código de la orden, el código del libro, su título y la cantidad de ejemplares a imprimir. Además nos pide que en todas las pantallas de la aplicación aparezca el logo de la tienda. También nos dice que tienen una base de datos de clientes, en la que de cada cliente se guarda su DNI, nombre, domicilio y tipo de cliente (básico, habitual o VIP). Comenta que el tipo de cliente se tiene en cuenta al generar una factura, que si el habitual se hace un descuento de un 5% y si es VIP un descuento del 10%. Nos pide que si quiere hacer un pedido un cliente que no existe en su base de datos, que el responsable de la tienda, y sólo él, pueda dar de alta al cliente desde la aplicación de pedidos. También nos aclara que en esa base de datos están ya los datos de todos los libros de la tienda, que cuando llegan libros la actualizan directamente con otra aplicación; y que de momento no hace falta en la aplicación de pedidos tener la opción de consultar los datos de los libros cuando los tienen que incluir en un pedido, que si tienen dudas sobre el código o título de un libro, lo miran desde la otra aplicación.*

*Por otra parte se discuten condiciones tecnológicas, sobre las cuales el responsable no tiene especiales preferencias, sólo que el tiempo de búsqueda de un libro sea breve, por ejemplo de menos de un segundo. Que la aplicación (de escritorio) funcione en Windows 10, que es el sistema que tienen los dos PCs nuevos de la tienda, y que hay otro PC en el que hay otras aplicaciones y la base de datos de clientes y libros. Además de una impresora por la que deben salir las facturas automáticamente cuando se generan. Todos estos dispositivos se conectan entre sí por una red local existente. Sobre seguridad, el responsable quiere que se acceda con el mismo usuario y contraseña que utilizan en el PC de la base de datos.*

Además del enunciado anterior, ya existente en la práctica 1, se ha realizado otra entrevista con el responsable de la tienda para aclarar algunas dudas del analista respecto al formato de algunos datos, obteniéndose la siguiente información:

*El responsable nos dice que el DNI de los clientes tiene 8 dígitos y una letra, que se obtiene a partir de los dígitos. Que el nombre de un cliente tiene nombre de pila, primer apellido y segundo apellido (opcional). Que el domicilio de los clientes, consta de calle, número, municipio, provincia y código postal. Que el código de un pedido debería empezar por la letra “P”. Que el código de un libro es su ISBN, con el formato estándar de 13 dígitos. Que el código de una factura debe empezar por la letra “F”. Que la fecha de emisión de una factura no puede ser anterior a la fecha del pedido asociado.*

*El responsable también nos explica que quiere que el empleado tenga los mismos privilegios de acceso que él a la aplicación si es un empleado de confianza, que en la base de datos de usuarios existe un campo que indica si un empleado es de confianza o no.*

**Se pide:**

1. Revisar el catálogo de requisitos y completarlo si falta algún requisito de datos.
2. Identificar las clases conceptuales que representan las entidades de información que manejará el software.
3. Identificar las relaciones entre las clases y dibujar el modelo conceptual de clases de la aplicación
4. Identificar los atributos de las clases y añadirlos al modelo conceptual de clases de la aplicación.

**1. Revisar el catálogo de requisitos y completarlo si falta algún requisito de datos.**

En la nueva entrevista se identifican nuevos requisitos de datos:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Id | Nombre | Descripción | Fuente | Tipo | Prioridad | Fecha | Autor |
| RD04.01 | DNI | El DNI de los clientes tiene 8 dígitos y una letra, que se obtiene a partir de los dígitos. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD04.02 | Nombre cliente | El nombre de un cliente tiene nombre de pila, primer apellido y segundo apellido (opcional). | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD04.03 | Domicilio cliente | El domicilio de los clientes, consta de calle, número, municipio, provincia y código postal. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD01.02 | Código pedido | El código de un pedido debería empezar por la letra “P”. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD02.02 | ISBN | El código de un libro es su ISBN, con el formato estándar de 13 dígitos. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD02.03 | Código factura | El código de una factura debe empezar por la letra “F”. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD02.04 | Fecha factura | La fecha de emisión de una factura no puede ser anterior a la fecha del pedido asociado. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |
| RD05.01 | Empleado confianza | Un empleado puede ser de confianza y tener los mismos privilegios que el responsable de la tienda. | Responsable | D | Alta | 07/02/20 | J.R. Hilera |

**2. Identificar las clases conceptuales que representan las entidades de información que manejará el software.**

Las clases y los atributos se obtienen de los sustantivos que aparecen en las entrevistas. Se elabora una lista y después se procede a descartar los que no representen información que manejará la aplicación, y a eliminar los que sean equivalentes a otros.

Se marcan los que sean candidatos a ser clases por tener otros sustantivos como atributos.

| **Sustantivo** | **Descartes.** | **Candidato a Clase** | **Candidato a atributo de la clase …** |
| --- | --- | --- | --- |
| Responsable |  | Responsable |  |
| ~~Tienda~~ | Descartar |  |  |
| Cliente |  | Cliente |  |
| Libro |  | Libro |  |
| Pedido |  | Pedido |  |
| Existencias de libro en la tienda |  |  | Libro |
| Código de pedido |  |  | Pedido |
| Fecha de pedido |  |  | Pedido |
| DNI del cliente en pedido |  |  | Pedido |
| Código de libro en pedido |  |  | Pedido |
| Cantidad de ejemplares de libro en pedido |  |  | Pedido |
| ~~Sistema~~ | Descartar |  |  |
| Factura de venta |  | Factura |  |
| Número de factura |  |  | Factura |
| Código de pedido en factura |  |  | Factura |
| DNI en factura |  |  | Factura |
| Fecha de emisión de factura |  |  | Factura |
| Nombre de cliente en factura |  |  | Factura |
| Domicilio de cliente en factura |  |  | Factura |
| Lista de libros en factura |  |  | Factura |
| Título de libro en factura |  |  | Factura |
| Precio de libro en factura |  |  | Factura |
| Precio final de la venta en factura |  |  | Factura |
| Orden de impresión |  | Orden de impresión |  |
| ~~Imprenta~~ | Descartar |  |  |
| ~~Títulos solicitados~~ | Descartar. Está desglosado en datos de una orden de impresión. |  | Orden de impresión |
| ~~Stock~~ | Descartar. Equivale a Existencias de libro en la tienda. |  |  |
| ~~Datos de un pedido~~ | Descartar. Equivale a Pedido. |  |  |
| ~~Pantallas~~ | Descartar |  |  |
| Pedido pendiente |  | Pedido pendiente |  |
| Empleado |  | Empleado |  |
| Título de un libro |  |  | Libro |
| ~~Aplicación~~ | Descartar |  |  |
| Código de orden de impresión |  |  | Orden de impresión |
| Código de libro (en orden de impresión) |  |  | Orden de impresión |
| Título de libro (en orden de impresión) |  |  | Orden de impresión |
| Cantidad de ejemplares a imprimir |  |  | Orden de impresión |
| Logo | Descartar |  |  |
| ~~Base de datos de clientes~~ | Descartar. Equivale a cliente. |  |  |
| Tipo de cliente |  |  | Cliente |
| Descuento a cliente |  |  | Cliente |
| ~~Duda~~ | Descartar |  |  |
| ~~Windows~~ | Descartar |  |  |
| ~~PC~~ | Descartar |  |  |
| ~~Impresora~~ | Descartar |  |  |
| ~~Dispositivo~~ | Descartar |  |  |
| ~~Red local~~ | Descartar |  |  |
| Nombre usuario |  |  | Usuario |
| Contraseña usuario |  |  | Usuario |
| Nombre de pila de cliente |  |  | Cliente |
| Primer apellido de cliente |  |  | Cliente |
| Segundo apellido de cliente |  |  | Cliente |
| Calle de cliente |  |  | Cliente |
| Número de calle de cliente |  |  | Cliente |
| Municipio de cliente |  |  | Cliente |
| Provincia de cliente |  |  | Cliente |
| Código postal de cliente |  |  | Cliente |
| Código de libro |  |  | Libro |
| ~~ISBN~~ | Descartado. Equivale a Código de libro |  |  |
| ~~Código de factura~~ | Descartado. Equivale a Número de factura. |  |  |
| Empleado de confianza |  |  | Empleado |

Por lo que los candidatos a clases y atributos son:

| **Clase** | **Atributos** |
| --- | --- |
| Pedido pendiente |  |
| Cliente | Calle de cliente |
| Código postal de cliente |
| Descuento a cliente |
| Municipio de cliente |
| Nombre de pila de cliente |
| Número de calle de cliente |
| Primer apellido de cliente |
| Provincia de cliente |
| Segundo apellido de cliente |
| Tipo de cliente |
| Factura | Código de pedido en factura |
| DNI en factura |
| Domicilio de cliente en factura |
| Fecha de emisión de factura |
| Lista de libros en factura |
| Nombre de cliente en factura |
| Número de factura |
| Precio de libro en factura |
| Precio final de la venta en factura |
| Título de libro en factura |
| Libro | Código de libro (es su ISBN) |
| Existencias del libro en la tienda |
| Título de libro |
| Precio de libro (no explícito en el enunciado, hay que saber leer entre líneas, cuando se habla de descuentos) |
| Orden de impresión | Código de orden de impresión |
| Código de libro |
| Título de libro |
| Cantidad de ejemplares a imprimir |
| Pedido | Código de libro en pedido |
| Cantidad de ejemplares de libro en pedido |
| Código de pedido |
| DNI del cliente en pedido |
| Fecha de pedido |
| Empleado | Empleado de confianza |
| Responsable |  |
| Usuario | Contraseña usuario |
| Nombre usuario |

No suele tener sentido que haya clases sin atributos, por lo que habrá que convertirlas en atributos de otras clases.

* La clase candidata “Pedido pendiente” no tiene atributos. Se puede convertir en un atributo de la clase pedido que indique el estado del pedido, y un posible valor para ese atributo podría ser “Pendiente”.
* En el caso de “Empleado” y “Responsable”, realmente son dos tipos de usuarios, pero en el caso de Empleado, además tiene un atributo más que indica si es de confianza, por lo que en este caso se puede aplicar la herencia, con una clase “Usuario” que contenga los atributos “Nombre” y “Contraseña”, y dos subclases “Responsable” y “Empleado”, añadiendo en este último caso como atributo si es de confianza. La clase “Responsable” no tiene atributos específicos, sólo los que hereda de “Usuario”, pero se mantiene en el modelo para diferenciar los dos tipos de usuarios.

Por tanto las clases que quedarían serían:

* Cliente
* Factura de Venta
* Libro
* Orden de Impresión
* Pedido
* Usuario
* Responsable
* Empleado

**3. Identificar las relaciones entre las clases y dibujar el modelo conceptual de clases de la aplicación**

Las relaciones se deben buscar en las frases del enunciado en las que aparezcan a la vez citados los nombres de dos clases. Por ejemplo, en la frase “el sistema debe generar la factura de venta correspondiente al pedido”, aparecen relacionadas “Pedido” y “Factura”. El término que las vincula (“correspondiente a”) puede ser el nombre de la relación en el diagrama de clases.

Hay que añadir la multiplicidad a las relaciones. En este caso, una factura corresponde a un solo pedido, y no puede haber facturas sin pedidos, por lo que la multiplicidad en el extremo “Pedido” de la relación sería “1..1” (que se puede simplificar con la notación “1”). En el otro extremo hay que indicar que un pedido puede tener como máximo una factura, pero podrían no tener ninguna factura asociada, si todavía está pendiente, por lo que en este caso sería 0..1.



Si se analizan el resto de relaciones, el diagrama quedaría como el siguiente:



En el enunciado pueden identificarse otras relaciones, pero son redundantes o se pueden derivar de las del diagrama. Por ejemplo hay una relación entre factura y libro, pero se puede eliminar, ya que esa relación ya existe implícitamente al estar una factura asociada a un pedido que a su vez incluye los libros, por lo que se puede llegar a los libros de la factura a través del pedido. No obstante, cuando se refine el diagrama de clases en la etapa de diseño, se podría decidir añadir algunas de estas relaciones por motivos de diseño, aunque no sean estrictamente necesarias.

**4. Identificar los atributos de las clases y añadirlos al modelo conceptual de clases de la aplicación.**

A continuación se añaden los atributos a las clases, identificados en un apartado anterior.

Hay algunos atributos que se puede descartar, si están repetidos en varias clases, en cuyo caso se podría llevar a una única clase y establecer una relación entre dicha clase y aquellas de las que se han eliminado dichos atributos. Esto ocurre en las clases Pedido y Factura, ya que en las dos aparecen atributos sobre libros y sobre clientes, que ya existen en las clases Libro y Cliente respectivamente. También en la clases Orden de impresión con los atributos código de libro y título de libro, que se pueden eliminar de la orden, ya que existen en Libro, y se puede llegar a ellos desde la relación.

En el diagrama se pueden indicar los tipos de los atributos. En esta etapa de análisis se puede utilizar texto libre, sin ajustarse a los tipos de algún lenguaje de programación.



En el diagrama no se han podido incluir como atributo en ninguna clase el número de ejemplares de cada libro que hay en un pedido, debido a que es una relación muchos a muchos, con multiplicidad \* en ambos extremos. Cuando esto ocurre, en UML se utilizan clases de asociación para contener dichos atributos.

