Diseño y desarrollo de un sistema de información de un concesionario: Sistema “NOMBRE”

Nombre del grupo: “Por Asignar”

Introducción

Hoy en día cualquier tipo de empresa necesita un sistema para gestionar el personal y la contabilidad de la misma, y la gran mayoría de ellas también involucran el control de un almacén de cualquier tipo, ya sea un almacén de productos de comida para un supermercado, un almacén de piezas y herramientas de una ferretería, o un almacén de vehículos de un concesionario. Es por ello que, al estar tan presente en la vida cotidiana este tipo de sistemas, hemos decidido diseñar y desarrollar un sistema de información apto para cualquier tipo de empresa que se encargue de gestionar un almacén.

Además, otro aspecto importante de cualquier empresa hoy en día es el análisis de datos de la misma, por lo que incluiremos un apartado dedicado a ello.

Precisamente, hemos decidido crear un sistema de información basado en la gestión de personal y control de almacén de un concesionario. Concretamente, nuestro sistema está ideado para gestionar de forma correcta el apartado de clientes, trabajadores, proveedores, almacén y contabilidad y análisis de datos de una empresa destinada a la venta de vehículos de segunda mano.

Vamos a diseñar este proyecto desde el punto de vista de que un administrador (ya puede ser un trabajador de la empresa o el propio jefe) gestione todas las funcionalidades del sistema, por lo que el mismo consta de un único agente externo: un empleado. Este se encargará de cualquier funcionalidad del sistema.

Objetivos

Establezcamos ahora los objetivos que queremos conseguir con nuestro sistema. El sistema deberá gestionar correctamente los clientes, trabajadores, proveedores, el almacén y la contabilidad y análisis de datos del concesionario.

En primer lugar, el sistema se encargará de gestionar los clientes de nuestro concesionario, dando lugar a un subsistema de gestión de clientes. A través del mismo permitiremos el tratamiento de los datos relativos a cada cliente, además de gestionar la compra de vehículos.

De forma análoga, el sistema se encargará de gestionar tanto los trabajadores como los proveedores del concesionario, dando lugar a dos subsistemas. El tratamiento de los datos de trabajadores y proveedores es exactamente el mismo que el de los clientes. A diferencia del primer subsistema, estos podrán añadir ventas a trabajadores y pedir vehículos a proveedores respectivamente.

Otro objetivo del sistema es el de administrar de forma correcta los vehículos del almacén, dando lugar a otro subsistema. Este se encarga de tramitar cualquier operación relativa a los vehículos del concesionario.

Por último, el sistema gestionará la contabilidad de la empresa y el análisis de datos de la misma, dando lugar al último subsistema. Con este objetivo se gestionan los salarios y el análisis de datos del concesionario.

Funcionalidades

En cuanto al primer subsistema, este almacenará los datos relativos a cada cliente. El empleado podrá tramitar tanto el alta como la baja de los clientes, además de todo lo relacionado con los datos de los mismos, incluyendo operaciones de modificación y consulta. La consulta de los clientes del concesionario se realizará según un criterio introducido por el empleado. Este se encargará también de gestionar y almacenar los datos de la compra de un vehículo del almacén por parte de un cliente. Esta comprá será efectiva siempre y cuando dispongamos de la cuenta bancaria del cliente.

De forma análoga, el segundo subsistema se encargará de gestionar y almacenar los datos de los trabajadores del concesionario. El tratamiento de los datos de trabajadores es el mismo que el de los clientes, el empleado podrá dar de alta, dar de baja, modificar y consultar los datos de trabajadores. A diferencia del primer subsistema, en lugar de gestionar la compra de un vehículo, este gestionará la apropiación de una venta por parte de un trabajador.

De nuevo, igual que en los dos subsistemas anteriores, el tercer subsistema se encargará de gestionar y almacenar los datos de proveedores. El tratamiento de los datos de proveedores es exactamente igual que en los anteriores, alta, baja, consulta y modificación. Además, este gestionará también la petición de un vehículo a un proveedor.

En cuanto a estos subsistemas, hemos considerado que tanto clientes y trabajadores como proveedores serán identificados por su DNI o CIF respectivamente, por lo que no podrán coexistir dos entidades en el sistema con el mismo DNI o CIF. Del mismo modo, no serán admitidos trabajadores que ya hayan sido contratados por la empresa.

Respecto al almacén, disponemos de un subsistema que lo gestiona. Este se encargará de registrar correctamente los vehículos proporcionados por los proveedores. Al igual que cualquier dato en nuestro sistema, los datos de los vehículos del almacén podrán ser consultados o modificados. Este subsistema también llevará a cabo la entrega de un vehículo a un cliente y el desguazamiento de un vehículo.

En relación al almacén en sí, hemos considerado que la entrada de los vehículos al almacén se realizará a través de la matrícula de los mismos, de modo que, al igual que en la realidad, no existan vehículos con la misma matrícula dentro del almacén. Es por ello que, cuando solicitamos un vehículo de un proveedor, al dar entrada al vehículo en el almacén le proporcionaremos una matrícula de origen Español. Por otro lado, no admitiremos vehículos con una antigüedad superior a los 15 años o con un kilometraje superior a los 200000 km, y tampoco admitiremos la entrada de vehículos los cuales ya hayan sido vendidos.

Por último, el sistema llevará a cabo la contabilidad del concesionario, dando lugar al último subsistema. En primer lugar, permitirá la gestión de los salarios, que registra cuándo se ha pagado a los trabajadores, la cantidad y si ha sido por transferencia o en efectivo. También se podrán calcular los ingresos brutos y los costes totales, y los resultados obtenidos, que se calculan a partir de los ingresos y de los costes. Para finalizar, el sistema también podrá consultar las estadísticas de los trabajadores.

Señalar que el resultado de confirmación o error de todas las operaciones descritas se mostrará con un mensaje indicativo por pantalla.

1. Clientes - Jesús Miguel Rojas Gómez:

1. Alta cliente [Añadir a la BDD los datos de un cliente nuevo]
2. Baja cliente [Eliminar de la BDD un cliente existente]
3. Listar clientes [Crear una lista con el nombre y las características indicadas de todos los clientes]
4. Modificar datos cliente [Modificar los datos de un cliente en la BDD]
5. Compra de vehículo [Añadir la compra de un vehículo del almacén a un cliente]

Crearemos un subsistema de información para almacenar los clientes de nuestro concesionario. De cada cliente, almacenaremos su nombre completo (en una serie de hasta 80 caracteres), su DNI (en una serie de 9 caracteres), un teléfono de contacto (en una serie de hasta 20 caracteres que pueden ser todos numéricos menos el primero que puede ser numérico o un signo +), un correo electrónico (en una serie de hasta 40 caracteres), una dirección (en una serie de hasta 100 caracteres) y una cuenta bancaria (en una serie de 24 caracteres).

Para **dar de alta un nuevo cliente**, el empleado deberá proporcionar obligatoriamente el nombre completo, DNI, número de teléfono y correo electrónico, y de forma opcional la dirección y la cuenta bancaria, datos que el sistema almacenará, confirmando la inserción o dando un error en caso de que el DNI del cliente ya exista en la BDD.

Para **dar de baja un cliente**, el empleado deberá proporcionar el DNI del cliente, confirmando el borrado o dando un error en caso de que el DNI proporcionado no coincida con ningún cliente existente. Hemos diseñado el borrado de clientes como una inserción en una tabla de clientes eliminados, con el fin de disponer de estos datos en caso de que otra funcionalidad del sistema los requiera.

Para **modificar los datos de un cliente**, el empleado deberá proporcionar el DNI del cliente y los datos que desea modificar, confirmando la modificación o dando un error en caso de que el DNI proporcionado no coincida con ningún cliente existente.

El sistema permitirá **mostrar un listado de clientes** con el criterio indicado a petición del empleado. Los criterios serán: 1. Ordenados alfabéticamente. 2. Clientes activos ordenados alfabéticamente. 3. Clientes eliminados ordenados alfabéticamente. 4. Clientes ordenados por número de vehículos comprados. 5. Clientes ordenados por número de vehículos comprados, a partir de x vehículos.

El sistema también permitirá que un cliente realice una **compra de un vehículo**. Para ello, el empleado deberá proporcionar el DNI de un cliente y la matrícula de un vehículo existente en el almacén, confirmando la compra o dando un error en caso de que el DNI del cliente no coincida con ningún cliente existente o en caso de que la matrícula no coincida con ningún vehículo de los existentes en el almacén. En caso de que el cliente indicado no tenga almacenada la cuenta bancaria, el empleado deberá proporcionar la cuenta bancaria de dicho cliente para poder realizar la compra.

## **RF1.1:** Dar de alta un cliente.

**Entrada:** Agente externo: empleado.

Acción: solicitar inserción.

Requisito de datos de entrada: **RDE1.1**.

**BD:** Requisito de datos de escritura: **RDW1.1**

Requisito de datos de lectura: **RDR1.1.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE1.1:** Datos de entrada de alta de cliente.

Nombre completo: Cadena de caracteres (80).

DNI: Cadena de caracteres (9).

Teléfono de contacto: Cadena de caracteres (20).

Correo electrónico: Cadena de caracteres (40).

Dirección: Cadena de caracteres (100).

Cuenta bancaria: Cadena de caracteres (24).

**RDR1.1:** Datos de cliente almacenados.

DNI: Cadena de caracteres (9).

**RDW1.1:** Datos almacenados de cliente.

Los mismos datos que **RDE1.1.**

## **RF1.2:** Dar de baja un cliente.

**Entrada:** Agente externo: empleado.

Acción: solicitar borrado.

Requisito de datos de entrada: **RDE1.2**.

**BD:** Requisito de datos de escritura: **RDW1.2.**

Requisito de datos de lectura: **RDR1.2.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE1.2:** Datos de entrada de DNI de cliente.

DNI: Cadena de caracteres (9).

**RDR1.2:** Datos de cliente almacenados.

DNI: Cadena de caracteres (9).

**RDW1.2:** Datos almacenados de cliente.

Los mismos datos que **RDE1.1.**

## **RF1.3:** Listar clientes.

**Entrada:** Agente externo: empleado.

Acción: solicitar listado.

Requisito de datos de entrada: **RDE1.3.**

**BD:** Requisito de datos de lectura: **RDR1.3.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: **RDS1.3.**

**RDE1.3:** Criterio seguido a la hora de listar los clientes.

Criterio: Entero.

**RDR1.3:** Datos de cliente almacenados.

Los mismos datos que **RDE1.1.**

**RDS1.3:** Listado de registros, cada uno de ellos con los mismos datos de **RDR1.3.**

## **RF1.4:** Modificar un cliente.

**Entrada:** Agente externo: empleado.

Acción: solicitar modificación.

Requisito de datos de entrada: **RDE1.4**.

**BD:** Requisito de datos de escritura: **RDW1.4.**

Requisito de datos de lectura: **RDR1.4.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE1.4:** Datos de entrada de modificar cliente.

Los mismos datos que **RDE1.1**

**RDR1.4:** Datos de cliente almacenados.

Los mismos datos que **RDE1.1.**

**RDW1.4:** Datos almacenados de cliente.

Los mismos datos que **RDE1.1.**

## **RF1.5:** Compra de un vehículo.

**Entrada:** Agente externo: empleado.

Acción: solicitar compra.

Requisito de datos de entrada: **RDE1.5**.

**BD:** Requisito de datos de escritura: **RDW1.5.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE1.5:** Datos de entrada de compra de un vehículo.

DNI: Cadena de caracteres (9).

Matrícula: Cadena de caracteres (8).

**RDR1.5:** Datos de entrada de compra de un vehículo.

DNI: Cadena de caracteres (9).

Cuenta bancaria: Cadena de caracteres (20).

Matrícula: Cadena de caracteres (8).

**RDW1.5:** Datos almacenados de compra de vehículo.

DNI cliente: Cadena de caracteres (9).

Matrícula: Cadena de caracteres (8).

Fecha de compra: Cadena de caracteres (10).

Precio: Entero.

## **RS1.1:** Un DNI corresponde a un único cliente.

**RF:** RF1.1, RF1.4.

**RD(s):** RDW1.1, RDE1.1, RDR1.1, RDW1.4.

**Descripción:** “Al insertar un nuevo cliente, si ya existe un cliente con el DNI proporcionado, no se inserta y se devuelve un error. Al modificar un cliente existente, en caso de que se proporcione un nuevo DNI, si ya existe un cliente con el DNI proporcionado, no se modifica y se devuelve un error”.

## **RS1.2:** Cuenta bancaria vacía al comprar un vehículo.

**RF:** RF1.5.

**RD(s):** RDW1.1, RDE1.1, RDR1.5.

**Descripción:** “Al realizar la compra de un vehículo, la cuenta bancaria del cliente no puede ser vacía”.

## **RS1.3:** Un vehículo solo se puede comprar una vez.

**RF:** RF1.5.

**RD(s):** RDE1.5.

**Descripción:** “Un vehículo solo puede ser comprado una vez. Esta restricción se controlará no permitiendo la entrada de vehículos al inventario que ya han sido vendidos ”.

2. Trabajadores - María Izquierdo:

1. Alta (Contratación) [Añadir a la BDD los datos de un trabajador nuevo]
2. Baja (Despido / Dimisión / Jubilación) [Marcar en la BDD al trabajador como que ya no trabaja en la empresa -> tipo de abandono de la empresa: jubilación, dimisión, despido…]
3. Modificar datos [Básicas + Cambio sueldo, posición en la empresa]
4. Añadir coche vendido [Añadir una venta al trabajador]
5. Consulta trabajadores [Crear una lista con el nombre y las características indicadas de todos los trabajadores]

Crearemos un subsistema de información para almacenar los trabajadores de nuestro concesionario. De cada trabajador almacenaremos su nombre completo (en una serie de hasta 80 caracteres), su DNI (en una serie de 9 caracteres), un teléfono de contacto (en una serie de hasta 20 caracteres que pueden ser todos numéricos menos el primero que puede ser numérico o un signo +), un correo electrónico (en una serie de hasta 40 caracteres), una dirección (en una serie de hasta 100 caracteres), una cuenta bancaria (en una serie de 20 caracteres), y el número de la seguridad social (en una serie de hasta 12 caracteres). También almacenaremos la posición en la empresa que ocupa el trabajador.

Para **dar de alta** a un trabajador serán necesarios todos los datos anteriormente especificados.

Para **dar de baja** a un trabajador, el empleado deberá proporcionar el DNI, confirmando el borrado o dando un error en el caso de que no coincida el DNI con el de ningún trabajador. En lugar de eliminar de los datos del trabajador, en el campo estado, que inicialmente está vacío, se completa el motivo de por qué ha sido dado de baja el trabajador.

Para **modificar los datos** de un trabajador, el empleado deberá proporcionar el DNI del trabajador y los datos que desea modificar, confirmando la modificación o dando un error en caso de que el DNI no coincida con ningún trabajador.

Para **añadir una venta a un trabajador** el empleado proporcionará la matrícula del vehículo y el DNI del trabajador asociado a la venta, confirmando la venta o dando un error en el caso de que no exista ningún vehículo registrado con esa matrícula. Un coche sólo puede ser vendido una vez.

El sistema también podrá mostrar un **listado de trabajadores** con los datos a petición del empleado.

## **RF2.1:** Dar de alta a un trabajador

**Entrada:** Agente externo: empleado.

Acción: solicitar inserción.

Requisito de datos de entrada: **RDE2.1**.

**BD:** Requisito de datos de escritura: **RDW2.1.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE2.1:** Datos de entrada de alta del trabajador.

Nombre completo: Cadena de caracteres (80).

DNI: Cadena de caracteres (9).

Teléfono de contacto: Cadena de caracteres (20).

Correo electrónico: Cadena de caracteres (40).

Dirección: Cadena de caracteres (100).

Cuenta bancaria: Cadena de caracteres (20).

Número seguridad social: Cadena de caracteres (12)

Salario: Entero

Estado: Cadena de caracteres (20)

**RDW2.1:** Datos almacenados del trabajador.

Los mismos datos que **RDE2.1.**

## **RF2.2:** Dar de baja a un trabajador

**Entrada:** Agente externo: empleado.

Acción: solicitar borrado.

Requisito de datos de entrada: **RDE2.2**.

**BD:** Requisito de datos de escritura: **RDW2.2.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE2.2:** Datos de entrada de DNI del trabajador.

DNI: Cadena de caracteres (9).

**RDW2.2:** Datos almacenados del trabajador.

**DNI: Cadena de caracteres (9)**

**Estado: Cadena de caracteres (20)**

## **RF2.3:** Listar trabajadores

**Entrada:** Agente externo: empleado.

Acción: solicitar listado.

Requisito de datos de entrada: **RDE2.3.**

**BD:** Requisito de datos de lectura: **RDR2.3.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: **RDS2.3.**

**RDE2.3:** Criterio seguido a la hora de listar los trabajadores.

Criterio: Entero.

**RDR2.3:** Datos del trabajador almacenados.

Los mismos datos que **RDW2.1.**

**RDS2.3:** Listado de registros, cada uno de ellos con los mismos datos de **RDR2.3.**

## **RF2.4:** Modificar un trabajador

**Entrada:** Agente externo: empleado.

Acción: solicitar modificación.

Requisito de datos de entrada: **RDE2.4**.

**BD:** Requisito de datos de escritura: **RDW2.4.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE2.4:** Datos de entrada de modificar trabajador.

Los mismos datos que **RDE2.1.**

**RDW2.4:** Datos almacenados de trabajador.

Los mismos datos que **RDE2.1.**

## **RF2.5:** Añadir coche vendido

**Entrada:** Agente externo: empleado.

Acción: confirmación venta.

Requisito de datos de entrada: **RDE2.5**.

**BD:** Requisito de datos de escritura: **RDW2.5.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE2.5:** Datos de entrada de añadir venta.

DNI: Cadena de caracteres (9).

Matrícula: Cadena de caracteres (8).

Fecha de venta: Cadena de caracteres (10)

**RDW2.5:**

DNI de trabajador: Cadena de caracteres (9).

Matrícula: Cadena de caracteres (7).

Fecha de venta: Cadena de caracteres (10)

## RS2.1: Un DNI corresponde a un único trabajador

RF: RF2.1, RF2.4

RD(s): RDW2.1, RDW2.4, RDE2.1, RDR2.1.

**Descripción:** “Al insertar un nuevo trabajador, si ya existe un trabajador con el DNI proporcionado, no se inserta y se devuelve un error. Al modificar un trabajador ya registrado, en caso de que se proporcione un nuevo DNI, si ya existe un trabajador con el DNI proporcionado, no se modifica y se devuelve un error”.

## RS2.2: Un trabajador dado de baja no puede volver a ser contratado

RF:RF2.2

RD(s): RDE2.2+RDW2.2

**Descripción:** “Si un trabajador ha sido dado de baja y se ha marcado como despido, dimisión o jubilación, este no puede volver a ser contratado”

3. Proveedores - Francisco Capilla:

1. Alta proveedor [Añadir a la BDD un nuevo proveedor]
2. Baja de proveedor actual [Marcar el proveedor como no disponible, es decir, que nuestra empresa no trabaja con ellos por el momento]
3. Consultar proveedores [Crear una lista con el nombre y las características indicadas de todos los proveedores]
4. Modificar proveedor [Básicas + “No disponible”]
5. Pedir vehículo [Pedir un vehículo a un proveedor]

Crearemos un subsistema de información para almacenar los proveedores de nuestro concesionario. De cada proveedor almacenaremos su CIF (cadena de 9 caracteres alfanumérica), nombre de la empresa (cadena de máximo 80 caracteres alfanumérica), teléfono (cadena de 9 caracteres numérica) y correo electrónico de contacto (cadena alfanumérica menor de 64 caracteres), dirección (cadena de caracteres), cuenta bancaria y lista de coches disponibles.

Para poder dar de **alta a un proveedor** serán necesarios todos los datos detallados anteriormente.

Para **pedir un vehículo** será necesario la matrícula del mismo como dato de entrada y que esa matrícula esté contenida en la lista de coches disponibles. La concesión es automática siempre que la matrícula esté contenida en dicha lista, y posteriormente se registraría en el almacén.

Para dar **de baja un proveedor**, se deberá proporcionar el nombre de la empresa y el CIF correspondiente, confirmando el borrado o dando un error en caso de que alguno de los datos proporcionados no coincida con ningún proveedor.

El sistema permitirá mostrar un **listado de proveedores** con todos sus datos a petición del empleado.

El sistema también permitirá **modificar los datos de los proveedores**.

Un empleado puede **pedir un vehículo** al proveedor, y este siempre se lo va a conceder, independientemente de si en el inventario este modelo está o no.

## **RF3.1:** Dar de alta un proveedor.

**Entrada:** Agente externo: empleado.

Acción: solicitar inserción.

Requisito de datos de entrada: **RDE3.1**.

**BD:** Requisito de datos de escritura: **RDW3.1.**

Requisito de datos de lectura: **RDR3.1.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE3.1:** Datos de entrada de alta de proveedor.

Nombre completo de la empresa: Cadena de caracteres (80).

CIF: Cadena de caracteres (9).

Teléfono de contacto: Cadena de caracteres (20).

Correo electrónico: Cadena de caracteres (40).

Dirección: Cadena de caracteres (100).

Cuenta bancaria: Cadena de caracteres (24).

**RDW3.1:** Datos almacenados del proveedor.

Los mismos datos que **RDE3.1.**

**RDR3.1**: Datos del proveedor almacenado.

CIF: Cadena de caracteres (9).

## **RF3.2:** Dar de baja un proveedor.

**Entrada:** Agente externo: empleado.

Acción: solicitar borrado.

Requisito de datos de entrada: **RDE3.2**.

Requisito de datos de lectura: **RDR3.2.**

**BD:** Requisito de datos de escritura: **RDW3.2.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE3.2:** Datos de entrada del CIF.

CIF: Cadena de caracteres (9).

**RDR3.2:** Datos del proveedor almacenado.

CIF: Cadena de caracteres (9).

**RDW3.2:** Datos almacenados del proveedor.

Los mismos datos que **RDW3.1.**

## **RF3.3:** Listar proveedores.

**Entrada:** Agente externo: empleado.

Acción: solicitar listado.

Requisito de datos de entrada: **RDE3.3.**

**BD:** Requisito de datos de lectura: **RDR3.3.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: **RDS3.3.**

**RDE3.3:** Criterio seguido a la hora de listar los proveedores.

Criterio: Entero.

**RDR3.3:** Datos del trabajador almacenados.

Los mismos datos que **RDW3.1.**

**RDS3.3:** Listado de registros, cada uno de ellos con los mismos datos de **RDR3.3.**

## **RF3.4:** Modificar un proveedor.

**Entrada:** Agente externo: empleado.

Acción: solicitar modificación.

Requisito de datos de entrada: **RDE3.4**.

**BD:** Requisito de datos de escritura: **RDW3.4.**

**Salida:** Agente externo: empleado.

Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE3.4:** Datos de entrada de modificar proveedor.

CIF: Cadena de caracteres (9).

Subconjunto de datos de **RDE3.1.**

**RDW3.4:** Datos almacenados del proveedor.

Los mismos datos que **RDE3.1.**

## **RF3.5:** Pedir vehículo.

**Entrada:** Agente externo: empleado.

Acción: solicitar pedido.

Requisito de datos de entrada: **RDE3.5.**

**BD:** Requisito de datos de lectura: **RDR3.5.**

**Salida:** Agente externo: empleado.

Acción: registro en inventario del coche.

Requisito de datos de salida: **RDS3.5.**

**RDE3.5:** Datos de proveedor y de vehículo a pedir.

Modelo: Cadena de caracteres (20).

Marca: Cadena de caracteres (10)

CIF: Cadena de caracteres (9)

**RDR3.5:** Datos de proveedor

CIF: Cadena de caracteres (9)

**RDS3.5:** Datos del vehículo.

Precio de compra: Entero

Fecha de compra: Fecha

Kilometraje: Entero

año de fabricación: Fecha

## RS3.1: Un CIF solo corresponde a un proveedor.

RF: RF3.1, RF3.4

RD(s): RDE3.1

**Descripción:** “Si ya había un proveedor registrado con el mismo CIF, no se inserta el nuevo proveedor y se devuelve un error.”

4. Almacén - Maribel Ruiz:

1. Registrar vehículo (Adquisición vehículo) [Solo doy de alta un vehículo cuando se realiza un pedido a un proveedor o la entrega del mismo por parte del cliente esté ya en el almacén]
2. Entregar un vehículo (Venta del vehículo) [Eliminar vehículo del almacén + Ganamos PVP por el vehículo]
3. Baja de un vehículo (Desguace) [Eliminar vehículo del almacén + Ganamos cierta cantidad por desguazarlo]
4. Modificar datos de un automóvil [Cambiar precio, descripción]
5. Consultar vehículos [Crear una lista con el nombre y las características indicadas de todos los vehículos]

Deseamos crear un sistema de información para un almacén que registre los automóviles que se encuentran en el mismo. De cada vehículo, almacenaremos su matrícula (en un serie de hasta ~~7~~ 8 caracteres alfanuméricos), su marca (en una serie de hasta 10 caracteres alfanuméricos), su modelo (en una serie de hasta 20 caracteres alfanuméricos), sus características técnicas (en una serie de hasta ~~40~~ 100 caracteres), su kilometraje (representado por un entero), su año de fabricación (representado por un entero)~~, su precio en euros (representado por un entero),~~ y su fecha de entrada (~~en una serie de hasta 10 caracteres~~ representada por un objeto de tipo fecha) ~~y su fecha de salida (en una serie de hasta 10 caracteres)~~.

Para **registrar** un automóvil en el almacén, el empleado deberá proporcionar la matrícula del mismo, su marca, su modelo, sus características técnicas, su kilometraje, su año de fabricación, su precio y su fecha de entrada, datos que el sistema almacenará, confirmando la inserción o dando un error. El empleado realizará el registro de un vehículo cuando se realiza un pedido a un proveedor o un cliente entrega un vehículo. Se presupone que todos los coches que se registran en el almacén están ya matriculados. Además, en caso de que se intente registrar un vehículo que ya aparece como insertado en la base de datos de los Vehículos del almacén (A.4), se devolverá un error. Igualmente, si se intenta registrar un vehículo con más de 15 años de antigüedad o 200000 kilómetros, se devolverá un error.

Para **entregar** un vehículo, el empleado deberá proporcionar la matrícula y la fecha de salida del mismo, confirmando el marcado como borrado o dando un error. Con la entrega del vehículo al cliente, el concesionario ganará el PVP del mismo. Se almacenará información sobre la entrega del vehículo (matrícula, fecha de salida y precio) para su posterior uso en el subsistema de contabilidad.

Para **desguazar** un vehículo, el empleado deberá proporcionar la matrícula y la fecha de salida del mismo, confirmando el marcado como borrado o dando un error. Los beneficios del concesionario se incrementarán también al mandar un vehículo a desguazar. Se almacenará información sobre el desguace del vehículo (matrícula, fecha de salida y precio) para su posterior uso en el subsistema de contabilidad.

El sistema permitirá **modificar los datos** de un vehículo, para lo cual el empleado deberá proporcionar la matrícula del mismo obligatoriamente y cualquier otro dato que se desee modificar opcionalmente.

El sistema permitirá **mostrar un listado** de vehículos con todos sus datos a petición del empleado, pudiendo éste indicar alguna condición de listado.

No se podrá volver a vender un vehículo que ya ha sido previamente vendido en el concesionario. Sin embargo, sí se podrá mandar al desguace un vehículo que haya sido vendido previamente en el concesionario.

## **RF4.1:** Registrar un vehículo en el almacén.

**Entrada:** Agente externo: empleado. Acción solicitar inserción. Requisito de datos de entrada **RDE4.1**.

**BD:** Requisito de datos de escritura **RDW4.1**.

Requisito de datos de lectura: **RDR4.1**.

**Salida:** Agente externo: empleado. Acción: confirmación resultado. Requisito de datos de salida: ninguno.

**RDE4.1**: Datos de entrada de registro de vehículo.

Matrícula: Cadena de caracteres (~~7~~ 8).

Marca: Cadena de caracteres (10).

Modelo: Cadena de caracteres (20).

Características técnicas: Cadena de caracteres (~~40~~ 100).

Kilometraje: Entero.

Año de fabricación: ~~Entero~~ Fecha.

~~Precio: Entero.~~

Fecha de entrada: ~~Cadena de caracteres (10)~~ Fecha.

~~Fecha de salida: Cadena de caracteres (10).~~

**RDR4.1**: Datos del vehículo almacenado.

Matrícula: Cadena de caracteres (~~7~~ 8).

**RDW4.1**: Datos almacenados del vehículo.

Los mismos que RDE4.1.

## **RF4.2:** Entregar un vehículo.

**Entrada:** Agente externo: empleado. Acción: solicitar entrega de vehículo.

Requisito de datos de entrada **RDE4.2**.

**BD:** Requisito de datos de escritura **RDW4.2**.

Requisito de datos de lectura: **RDR4.2**.

**Salida:** Agente externo: empleado. Acción: confirmación resultado.

Requisito de datos de salida: ninguno.

**RDE4.2**: Datos de entrada de entrega de vehículo.

Matrícula: Cadena de caracteres (~~7~~ 8).

Fecha de salida: ~~Cadena de caracteres (10)~~ Fecha.

**RDR4.2**: Datos del vehículo almacenado.

Matrícula: Cadena de caracteres (~~7~~ 8).

**RDW4.2**: Datos almacenados del vehículo.

Matrícula: Cadena de caracteres (8).

Fecha de salida: Fecha.

Precio: Entero.

## **RF4.3:** Desguazar un vehículo.

**Entrada:** Agente externo: empleado. Acción: solicitar desguace de vehículo.

Requisito de datos de entrada **RDE4.3**.

**BD:** Requisito de datos de escritura **RDW4.3**.

Requisito de datos de lectura: **RDR4.3**.

**Salida:** Agente externo: empleado. Acción: confirmación resultado. Requisito de datos de salida: ninguno.

**RDE4.3**: Datos de entrada de desguace de vehículo.

Matrícula: Cadena de caracteres (~~7~~ 8).

Fecha de salida: ~~Cadena de caracteres (10)~~ Fecha.

**RDR4.3**: Datos del vehículo almacenado.

Matrícula: Cadena de caracteres (~~7~~ 8).

**RDW4.3**: Datos almacenados del vehículo.

Matrícula: Cadena de caracteres (8).

Fecha de salida: Fecha.

Precio: Entero.

## **RF4.4:** Modificar los datos de un vehículo.

**Entrada:** Agente externo: empleado. Acción: solicitar modificación de los datos de un vehículo. Requisito de datos de entrada **RDE4.4**.

**BD:** Requisito de datos de escritura **RDW4.4**.

Requisito de datos de lectura: **RDR4.4**.

**Salida:** Agente externo: empleado. Acción: confirmación resultado. Requisito de datos de salida: ninguno.

**RDE4.4**: Datos de entrada de modificación de los datos de un vehículo.

Los mismos que RDE4.1.

**RDR4.4**: Datos del vehículo almacenado.

Matrícula: Cadena de caracteres (~~7~~ 8).

**RDW4.4**: Datos almacenados del vehículo.

Los mismos que RDE4.1.

## **RF4.5:** Mostrar un listado de los vehículos que posee el almacén.

**Entrada:** Agente externo: empleado. Acción: solicitar listado de vehículos. Requisito de datos de entrada: **RDE4.5**.

**BD:** Requisito de datos de lectura **RDR4.5**.

**Salida:** Agente externo: empleado. Acción: confirmación resultado. Requisito de datos de salida: **RDS4.5**.

**RDE4.5**: Criterio de listado de los datos de los vehículos.

Criterio de listado: Entero.

**RDR4.5**: Datos del vehículo almacenado.

Los mismos datos que RDE4.1.

**RDS4.5**: Listado de registros, cada uno de ellos con los mismos datos de RDR4.5 y en el orden indicado por el criterio de listado.

## **RS4.1:** Una matrícula corresponde a un único vehículo.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si ya había un vehículo registrado con la misma matrícula, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.2:** ~~Un vehículo no excede los 15 años de antigüedad~~ ni tiene un kilometraje superior a 200000.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si el vehículo a insertar supera los 15 años de antigüedad o tiene un kilometraje superior a 200000, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.3:** La marca de un vehículo no puede ser nula.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si el vehículo a insertar no dispone de marca, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.4:** El modelo de un vehículo no puede ser nulo.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si el vehículo a insertar no dispone de modelo, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.5:** El kilometraje de un vehículo no puede estar vacío.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si el vehículo a insertar no dispone de kilometraje, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.6:** El año de fabricación de un vehículo no puede estar vacío.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si el vehículo a insertar no dispone de su año de fabricación, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.7:** Un vehículo que esté actualmente en el inventario debe disponer de características técnicas.

RF: RF4.1

RD(s): RDW4.1 + RDE4.1 + RDR4.1

**Descripción:** “Si el vehículo a insertar no dispone de características técnicas, no se inserta el nuevo vehículo y se devuelve un error.”

## **RS4.8:** El precio por el cual se ha vendido un vehículo no puede ser nulo y debe ser un número positivo o cero.

RF: RF4.2

RD(s): RDW4.2 + RDE4.2 + RDR4.2

**Descripción:** “Si un vehículo a vender no dispone de precio no nulo y mayor o igual que cero, no se vende vehículo y se devuelve un error.”

## **RS4.9:** El precio por el cual se ha desguazado un vehículo no puede ser nulo y debe ser un número positivo o cero.

RF: RF4.3

RD(s): RDW4.3 + RDE4.3 + RDR4.3

**Descripción:** “Si un vehículo a desguazar no dispone de precio no nulo y mayor o igual que cero, no se desguaza el vehículo y se devuelve un error.”

5. Contabilidad - José Luis Guidú

1. Pago de salarios [Pagar a los trabajadores y marcar en la BDD fecha, cantidad, efectivo, transferencia]
2. Ingresos brutos [Cálculo de todos los ingresos (coches vendidos + coches desguazados)]
3. Costes totales [Cálculo de todos los costes (costes fijos + coches comprados a proveedores + salarios)]
4. Resultados [IB - CT mes a mes y puedes ver los resultados de un periodo]
5. Estadística trabajadores [Clasificación de trabajadores por mes, año, etc]

Crearemos un subsistema de información para almacenar los datos referentes a la contabilidad de la empresa. También será el encargado de gestionar la interpretación de dichos datos. La base de datos almacenará los ingresos brutos de cada mes (en un número valorado en euros), los costes totales de cada mes (en un número valorado en euros), los beneficios/resultados obtenidos de cada mes (en un número valorado en euros) y de cada trabajador se guardarán los pagos con una gestión de las fechas, salario y si el pago se realizó en efectivo o por transferencia bancaria. El sistema también permitirá mostrar un listado ordenado según una clasificación de los trabajadores que venden más vehículos o ganen más dinero para la empresa.

Los ingresos brutos se calcularán obteniendo de la base de datos del almacén el precio y la fecha de salida de todos los vehículos vendidos. Se comparará la fecha de salida con el intervalo solicitado por el agente externo y se sumarán todos los que correspondan.

Los costes totales se calcularán obteniendo los costes fijos de la empresa [BD de contabilidad], el precio de los coches comprados a los proveedores en el intervalo solicitado [este dato se obtiene mirando en la BD del almacén; la matrícula y la fecha de entrada de los coches, buscamos en la BD de los proveedores dichas matrículas y sacamos sus precios proveedor], y los salarios pagados en dicho intervalo [de la BD de contabilidad sumamos los salarios pagados en dicho intervalo].

Los resultados se calcularán haciendo la diferencia entre los ingresos brutos y los costes totales del periodo especificado.

La estadística de los trabajadores se obtendrá leyendo de la BD todos los DNI de los trabajadores en plantilla, para cada trabajador obtenemos los precios de los coches que han vendido en el periodo solicitado.

## RF5.1: Pago de Salarios.

Entrada: Agente externo: empleado. Acción: solicitar inserción. RDE5.1

BD:

Requisito de datos de lectura RDR5.1

Requisito de datos de escritura RDW5.1

Salida: Agente externo: empleado. Acción: confirmación resultado

Requisitos de datos de salida: ninguno.

RDR5.1: Datos almacenados sobre el trabajador.

DNI: Cadena de caracteres (9).

Salario: Entero

RDW5.1: Datos almacenados sobre el pago al trabajador en la BD Contabilidad.

DNI: Cadena de caracteres (9).

Salario: Entero

FechaPago: Cadena de caracteres (10).

/\*COMENTARIO:

En la tabla RegistroSalarios se guardan el DNI, Salario y Fecha de Pago de este para que sirva de histórico, saber si se le ha pagado ya a un trabajador y poder hacer consultas

\*/

## RF5.2: Ingresos brutos

Entrada: Agente externo: empleado. Acción: solicita ingresos brutos.

Requisito de datos de entrada RDE5.2

BD: Requisito de datos de lectura RDR5.2

Salida: Agente externo: empleado. Acción: confirmación resultado.

Requisito de datos de salida: RDS5.2

RDE5.2: Periodo del que vamos a obtener los ingresos brutos

Fecha Inicio: Cadena de caracteres (10)

Fecha Fin: Cadena de caracteres (10)

RDR5.2: Datos de los ingresos de todos los coches vendidos y coches desguazados en el periodo temporal indicado en RDE5.2:

Precio y Fecha de Salida de los coches vendidos en el periodo RDE5.2.

Precio y Fecha de Salida de los coches desguazados en el periodo RDE5.2.

RDS5.2: Dato de la suma de todos los ingresos del periodo (precio coches vendidos + precio coches desguazados): Entero

/\*COMENTARIO:

Hay tablas diferentes para coches desguazados y coches vendidos (gestión almacén)

\*/

## RF5.3: Costes totales

Entrada: Agente externo: empleado. Acción: solicita costes totales.

Requisito de datos de entrada RDE5.3

BD: Requisito de datos de lectura RDR5.3

Salida: Agente externo: empleado. Acción: confirmación resultado.

Requisito de datos de salida: RDS5.3

RDE5.3: Periodo del que vamos a obtener los costes totales

Los mismos que RDE5.2

RDR5.3: Datos de costes:

Matrícula: Cadena de caracteres (8).

Fecha Entrada: Cadena de caracteres (10).

Precio Proveedor: Entero.

Costes Fijos: Entero.

Salario: Entero.

Fecha de Pago: Cadena de caracteres (10).

RDS5.3: Dato de la suma de todos los costes del periodo (costes fijos + precio proveedor de los coches que entraron al almacén en ese periodo + salarios pagados): Entero

/\*COMENTARIO:

El Precio Proveedor no lo encuentro en el esquema externo de proveedor.

Costes Fijos:

Crear una tabla que incluya el datito de los costes fijos

Costes Coche:

Cogemos de Almacén las matrículas de los coches con FechaEntrada en rango del periodo.

En Proveedores buscamos las matrículas y cogemos su precio de compra.

Costes Salarios:

Cogemos de Trabajadores el sueldo de los empleados cuya FechaPago esté dentro rango del periodo.

\*/

## RF5.4: Resultados

Entrada: Agente externo: empleado. Acción: solicita resultados del periodo.

Requisito de datos de entrada RDE5.4

BD: Requisito de datos de lectura RDR5.4

Salida: Agente externo: empleado. Acción: confirmación resultado.

Requisito de datos de salida: RDS5.4

RDE5.4: Periodo del que vamos a obtener los resultados

Los mismos que RDE5.2

RDR5.4: Datos de los ingresos brutos; los mismos que los de RDR5.2

Los datos de los costes totales; los mismos que los de RDR5.3.

Ambos del periodo temporal indicado en RDE5.4

RDS5.4: Dato de la diferencia de los beneficios y los costes totales del periodo: Entero

## RF5.5: Estadística trabajadores

Entrada: Agente externo: empleado. Acción: solicita estadísticas de los trabajadores.

Requisito de datos de entrada RDE5.5

BD: Requisito de datos de lectura RDR5.5

Salida: Agente externo: empleado. Acción: confirmación resultado.

Requisito de datos de salida: RDS5.5

RDE5.5: Periodo del que vamos a obtener las estadísticas

Los mismos que RDE5.2

RDR5.5: Datos de los trabajadores y sus ventas del periodo temporal indicado en RDE5.4:

DNI: Cadena de caracteres (9).

Fecha de Salida: Cadena de caracteres (10).

Matrícula: Cadena de caracteres (8).

Precio: Entero.

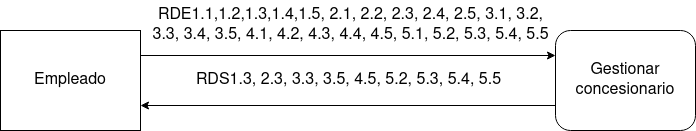
RDS5.5: Listado de tuplas (DNI del trabajador, su nombre y apellidos junto con la suma de los ingresos que ha obtenido en el periodo especificado) ordenado según la suma de ingresos para la empresa.

/\*COMENTARIO:

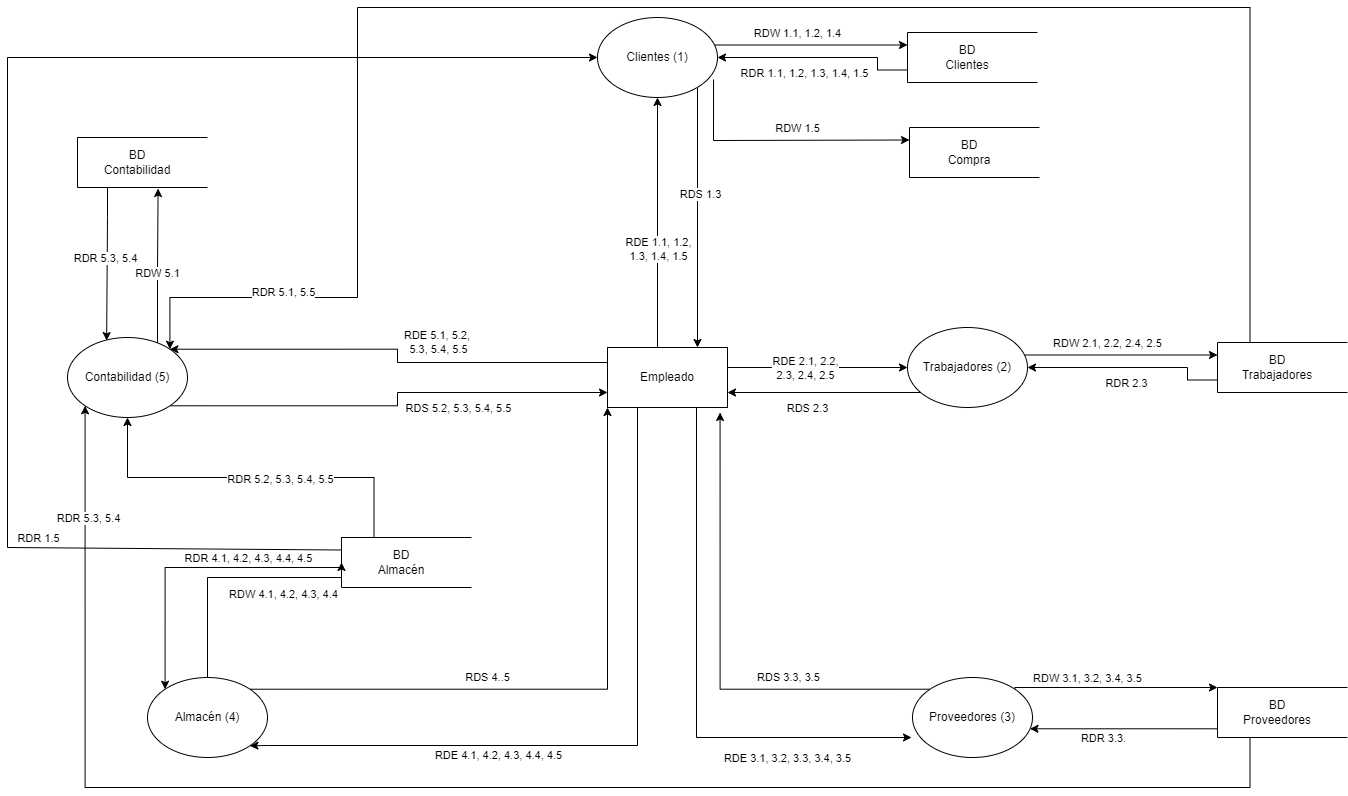
Se cogen los dni de los trabajadores que hayan vendido un coche en el periodo seleccionado, se busca la matrícula asociada en su tabla de coches vendidos y se obtiene el precio.

\*/

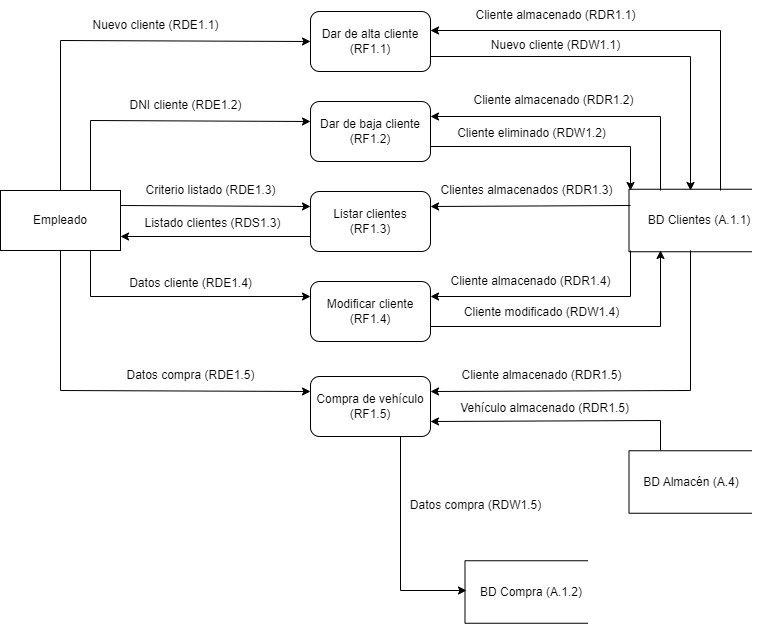
DFD Esquema de Caja Negra



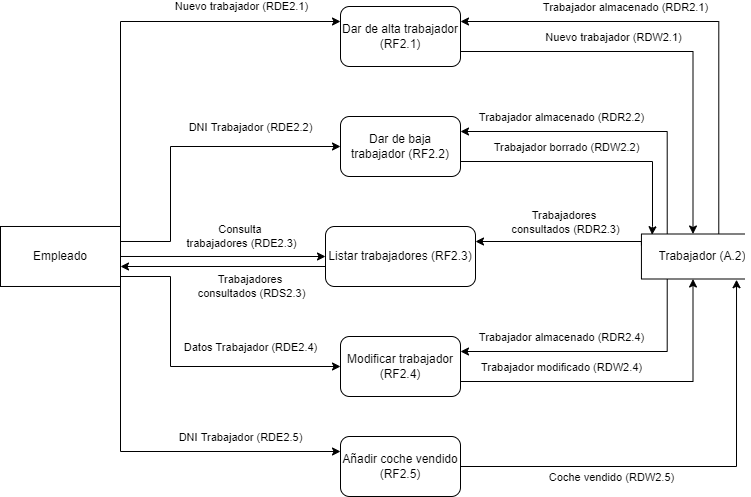
DFD Armazón



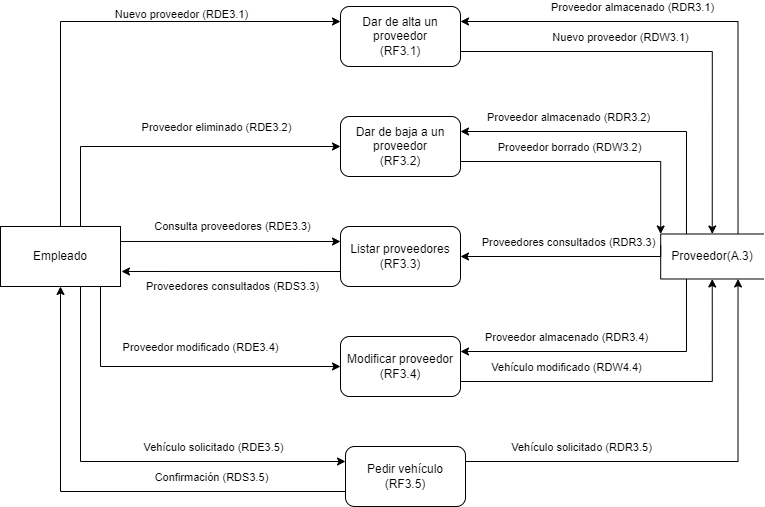
DFD Subsistema Clientes



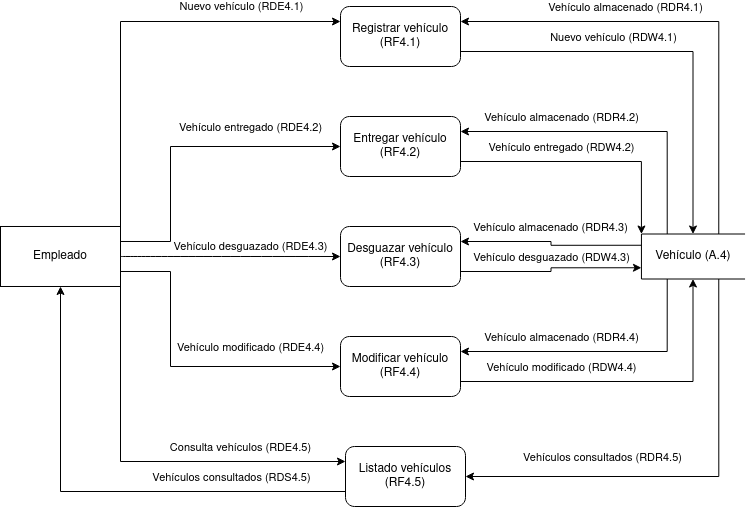
DFD Subsistema Trabajadores



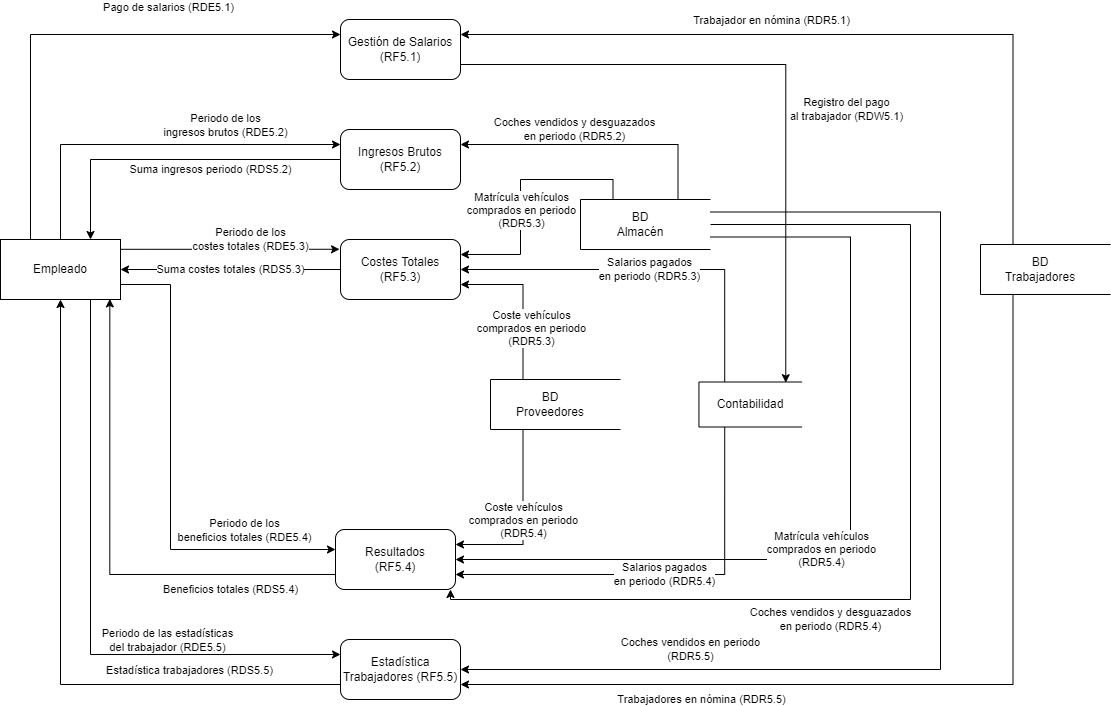
DFD Subsistema Proveedores



DFD Subsistema Almacén



DFD Subsistema Contabilidad



Correcciones de la práctica 1.

# Subsistema Clientes:

Corrección, cambio de cuenta bancaria por una serie de 24 dígitos en lugar de 20.

Añadido criterios de listado. Estos son:

1. Ordenados alfabéticamente
2. Ordenados alfabéticamente clientes activos
3. Ordenados alfabéticamente clientes eliminados
4. Ordenados por número de vehículos comprados
5. Ordenados por número de vehículos comprados, a partir de x vehículos comprados.

# Subsistema Trabajadores:

1. En el RF2.5 añado un nuevo requisito de entrada y de escritura, fecha de venta.
2. Completo la descripción, añadiendo más detalle del RF2.2.
3. Del esquema externo de añadir coche vendido elimino el atributo fecha de venta de la relación vende

# Subsistema Proveedores:

Se han modificado tanto RDE 3.5 como RDR 3.5 y RDS 3.5.

# Subsistema Almacén:

Se ha modificado la descripción, especificando con más detalle los requisitos funcionales RF4.2 y RF4.3.

Se ha modificado el requisito de datos de entrada RDE4.1. En concreto, se ha eliminado de los datos de entrada de registro de vehículo el precio y la fecha de salida. Además, el Año de fabricación es ahora de tipo Fecha.

Se han modificado también los requisitos de datos RDR4.1, RDE4.2, RDR4.2, RDW4.2, RDE4.3, RDR4.3, RDW4.3 y RDR4.4.

Se han añadido más restricciones semánticas (RS4.3, RS4.4, RS4.5, RS4.6, RS4.7, RS4.8, RS4.9).

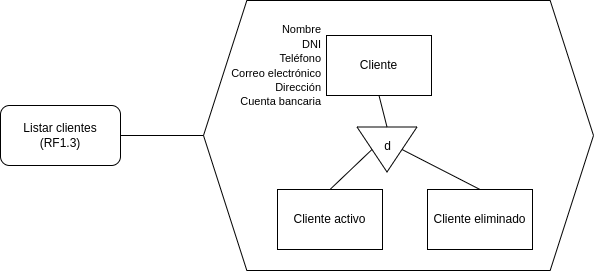
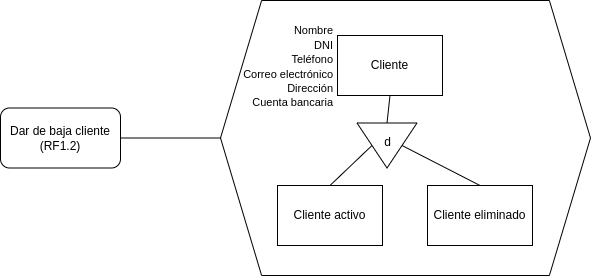
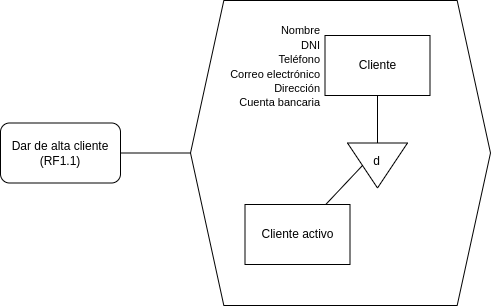
# Subsistema Contabilidad:

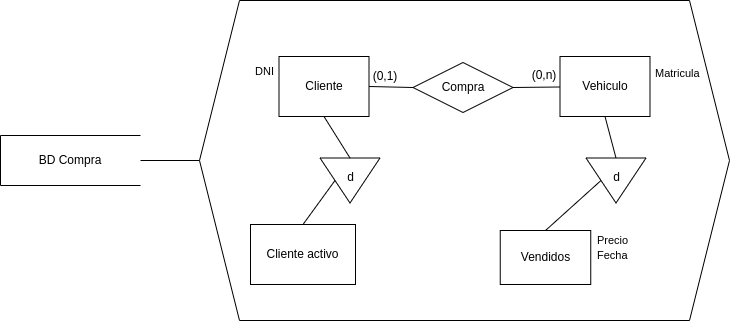
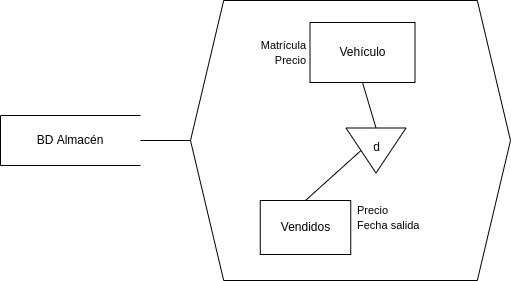
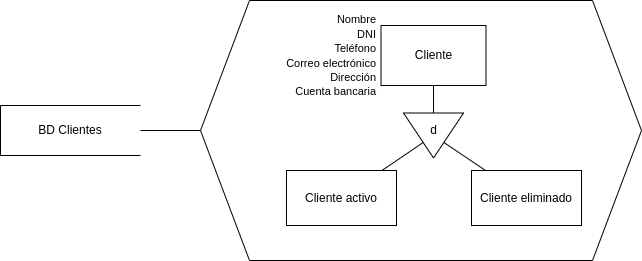
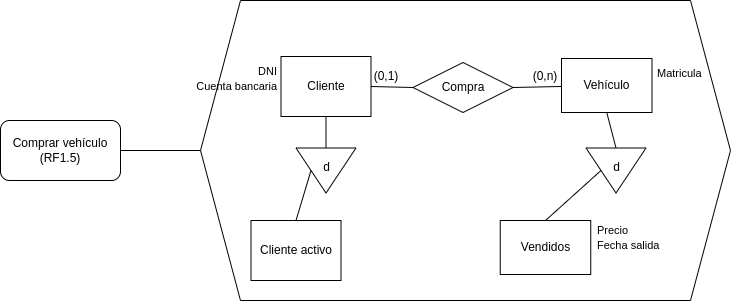
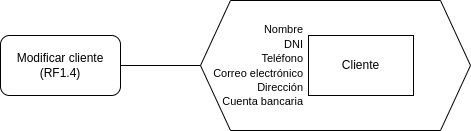
En el proceso 5.1 se ha cambiado el nombre del subsistema “Gestión de salarios” por “Pago de salarios” y se ha borrado la opción del método de pago del RDW5.1.

En el proceso 5.5 se ha especificado la salida más detalladamente (DNI y SUMA DE LOS INGRESOS).

Esquemas Externos de nivel 1

# Subsistema Clientes



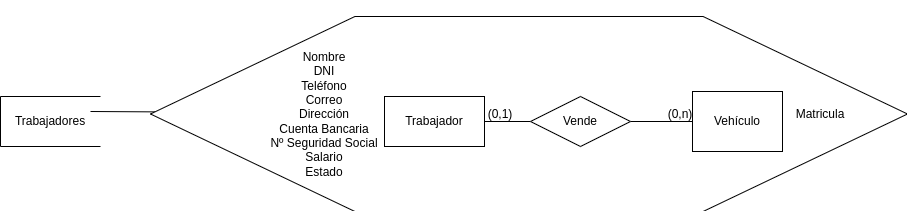


# Subsistema Trabajadores

# 

# 

# 

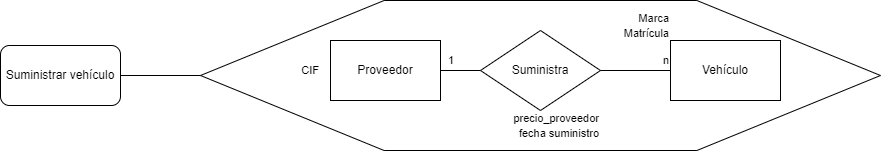
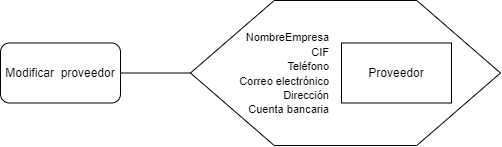
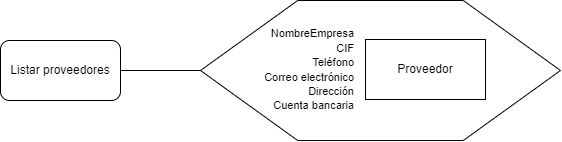


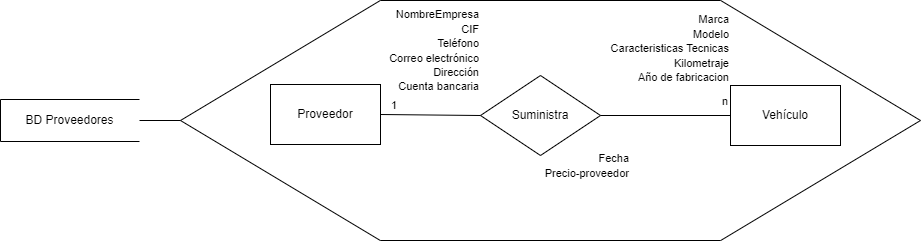
# 

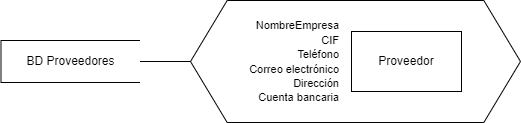
# 

# Subsistema Proveedores

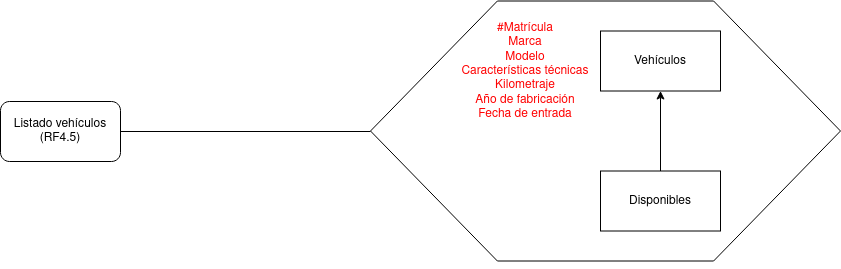
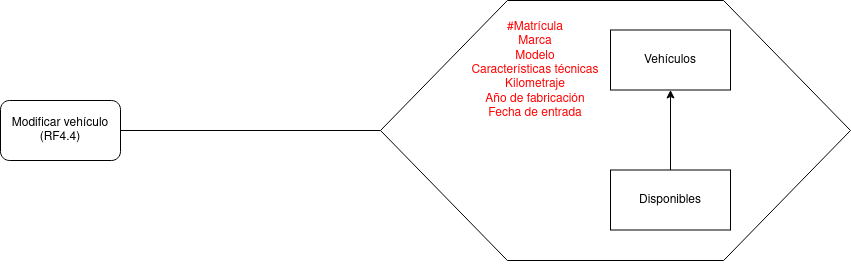
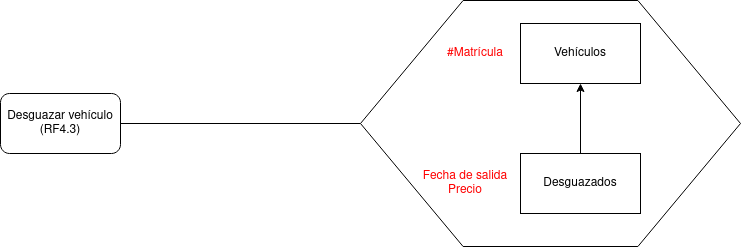
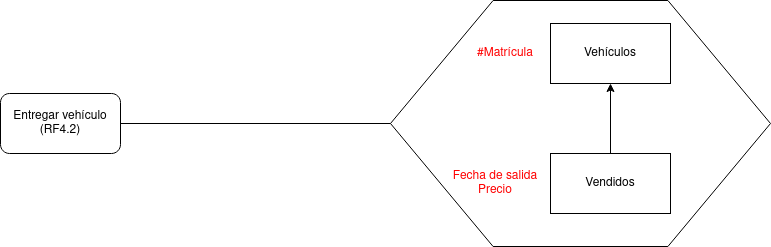
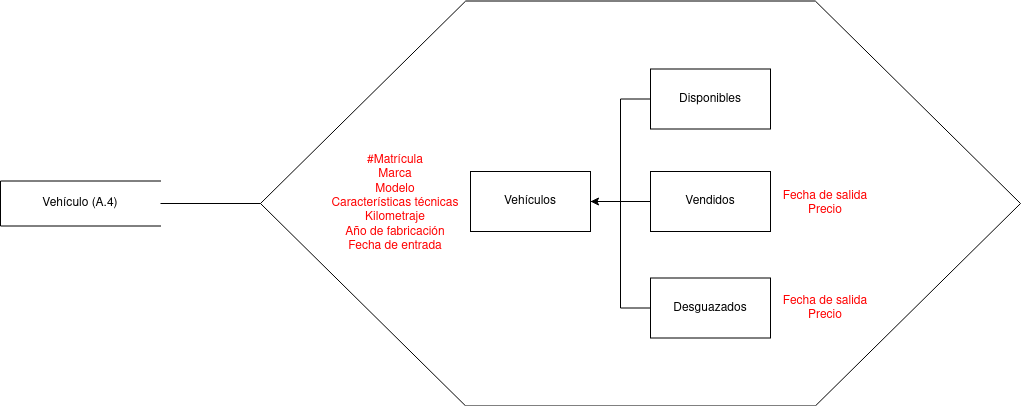




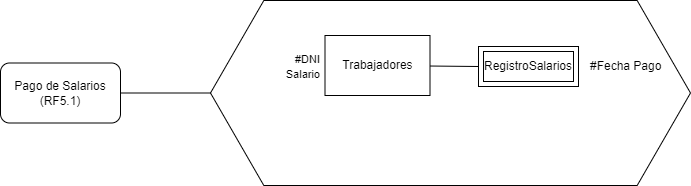


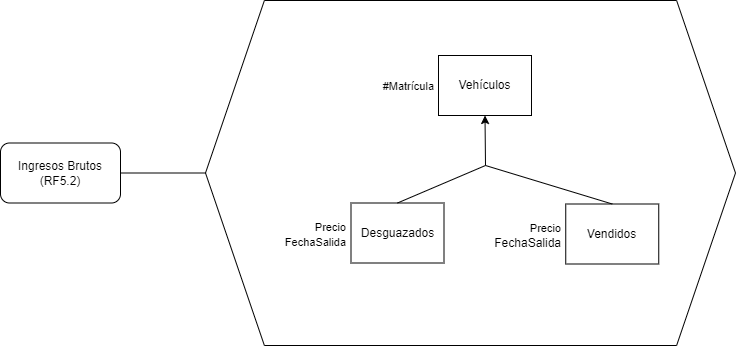


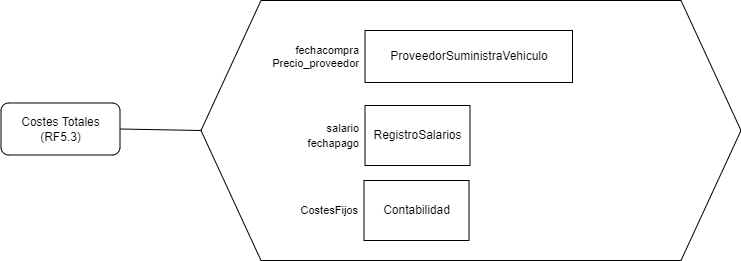
# Subsistema Almacén

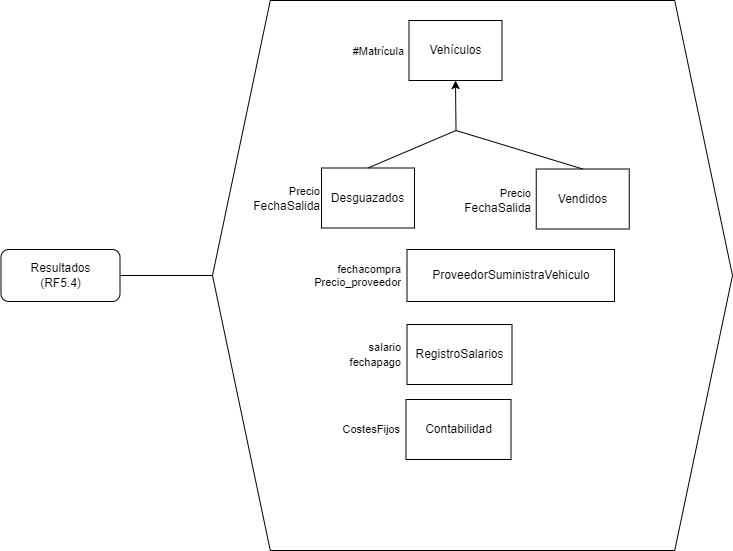
A continuación se muestra el esquema externo del almacén Vehículo. Se muestra un supertipo con varios subtipos (especialización) en correspondencia con lo expuesto en la descripción proporcionada en la Práctica 1.

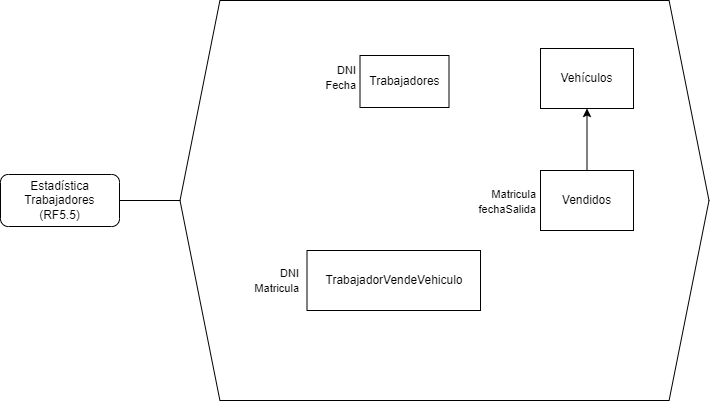
# Subsistema Contabilidad

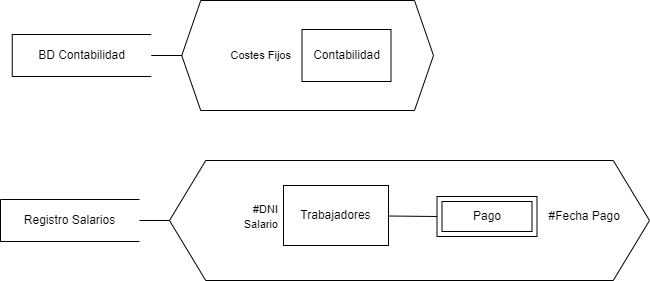


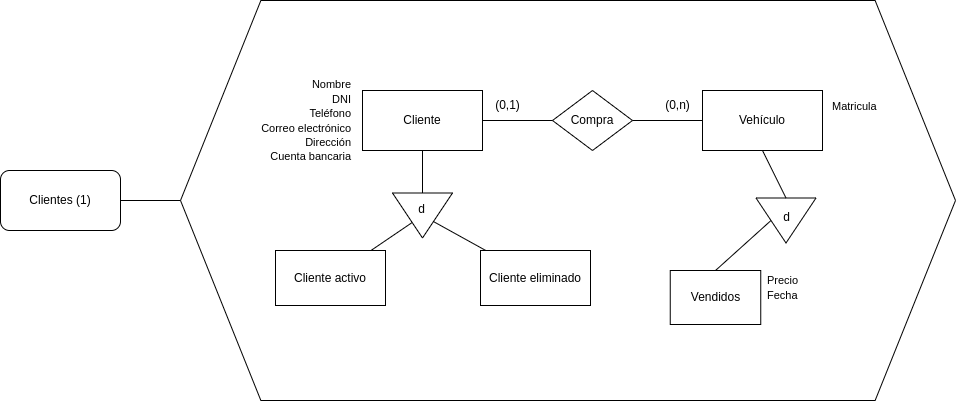


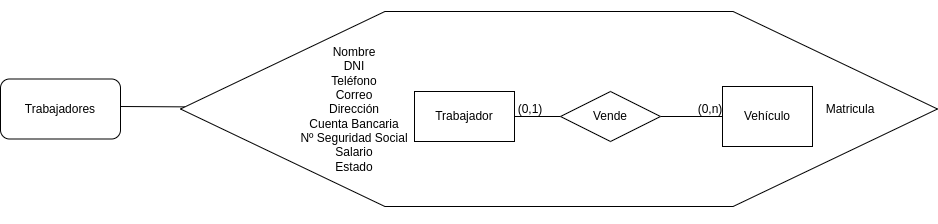


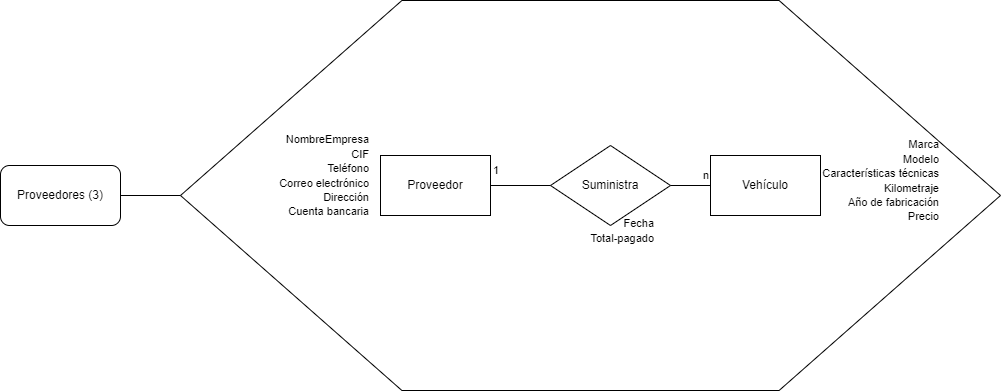






Esquemas Externos de nivel 0





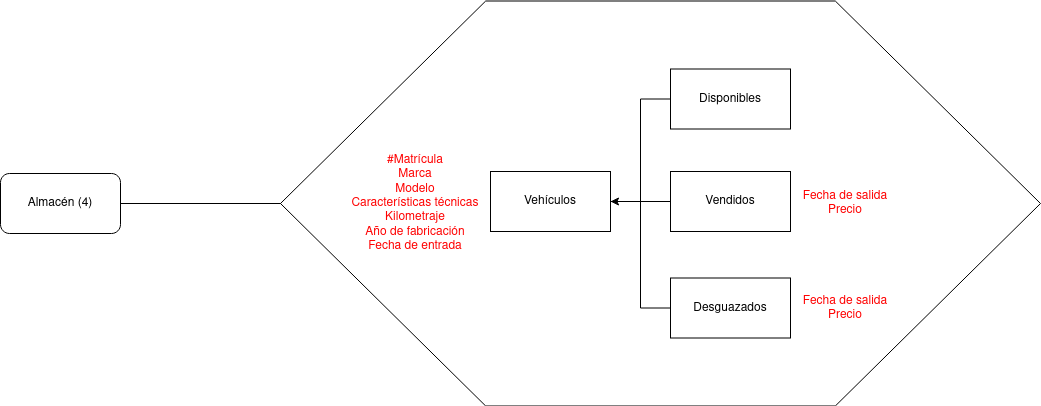
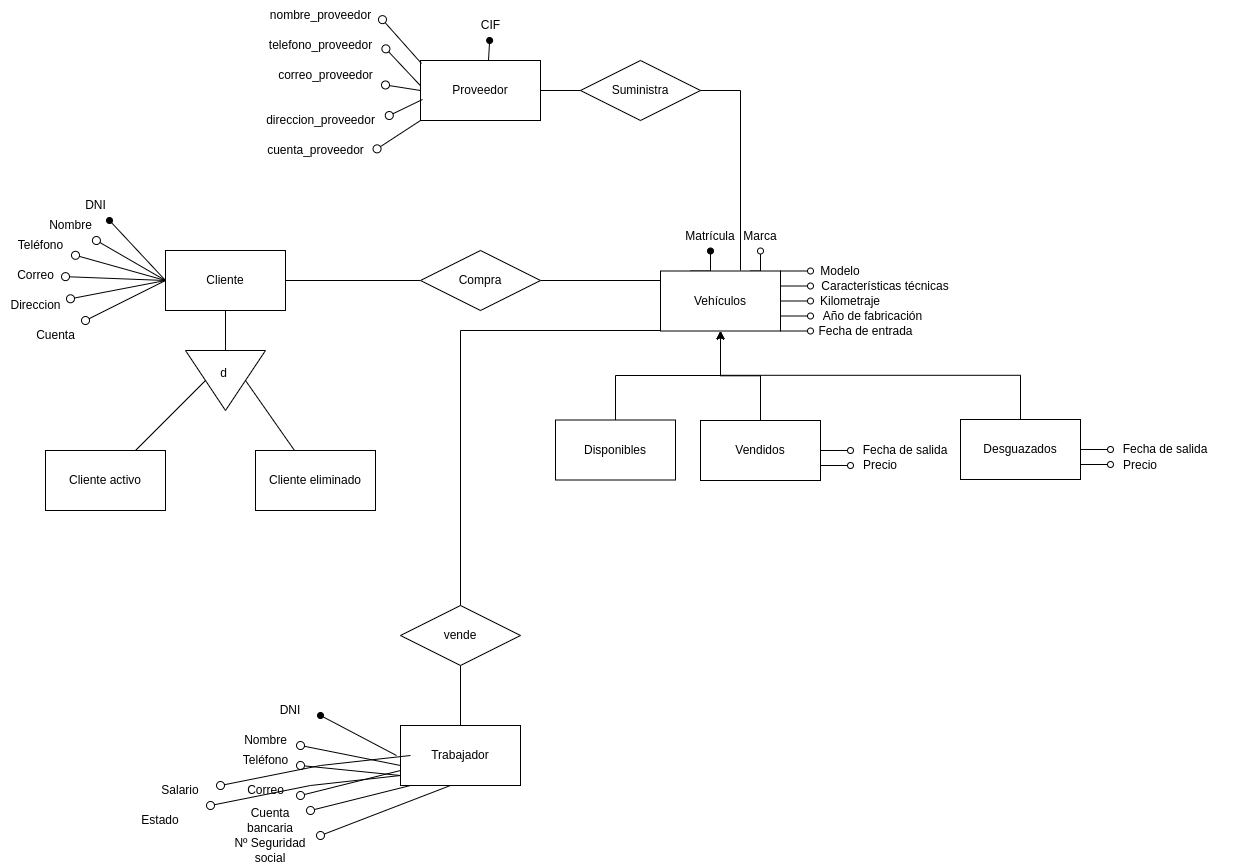
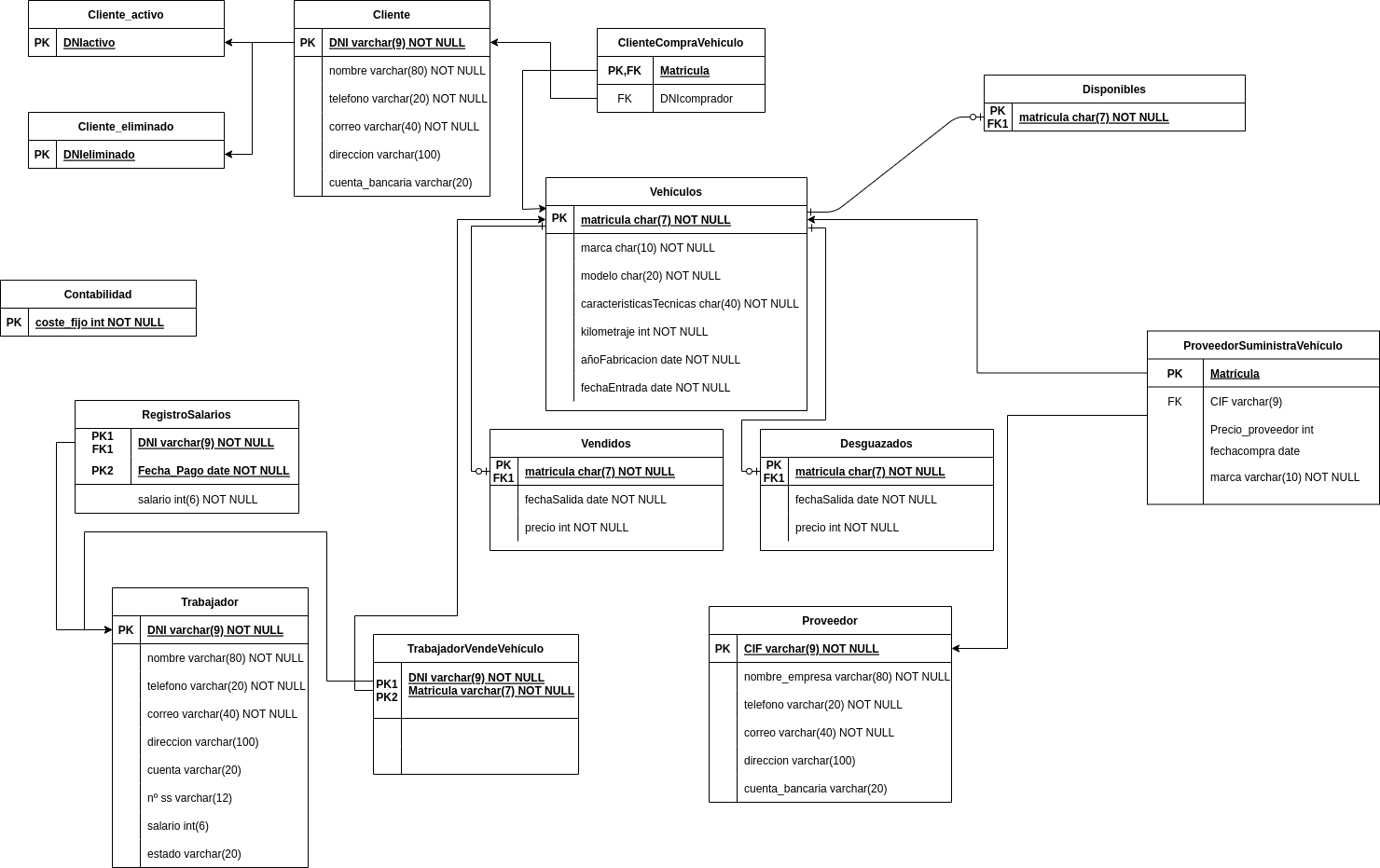


Diagrama E/R

Paso a tablas del modelo E/R



Normalización de las relaciones

# Subsistema Clientes

## Normalización de la relación Clientes

Cliente(DNI, nombre, telefono, correo, direccion, cuenta\_bancaria)

Cliente\_activo(DNI)

Cliente\_eliminado(DNI)

Dependencias funcionales no triviales:

DF = {

DNI → nombre | telefono | correo | direccion | cuenta\_bancaria,

}

La unica clave candidata es DNI, que es clave primaria. Como verifica la unicidad, DNI es superclave y la tabla está en FNBC.

Por otro lado, Cliente\_activo y Cliente\_eliminado ya están en FNBC.

## Normalización de la relación ClienteCompraVehiculo

ClienteCompraVehiculo(Matricula, DNI, Fecha, Precio)

Dependencias funcionales no triviales:

F = {

Matricula → DNI | Fecha | Precio

}

La única clave candidata es Matricula, que es clave primaria. Como verifica la unicidad, Matricula es superclave y la tabla está en FNBC.

# Subsistema Trabajadores

## Normalización de la relación Trabajadores

Trabajadores(DNI,nombre,teléfono,correo,dirección,cuenta\_bancaria,num\_seguridadsocial, salario,estado)

Dependencias funcionales:

* DNI → nombre | telefono | correo | direccion | cuenta\_bancaria | num\_seguridadsocial | salario | estado
* correo → DNI | nombre | telefono | direccion | cuenta\_bancaria | num\_seguridadsocial | salario | estado
* num\_seguridadsocial →DNI | nombre | telefono | correo | direccion | cuenta\_bancaria | salario | estado

Si dos tuplas tienen el mismo DNI, tienen que tener el mismo nombre, correo, direccion, cuenta\_bancaria, num\_seguridadsocial, salario y estado.

Si dos tuplas tienen el mismo num\_seguridadsocial, tienen que tener el mismo DNI, nombre, correo, direccion, cuenta\_bancaria, salario y estado.

Si dos tuplas tienen el mismo correo, tienen que tener el mismo DNI, nombre, direccion, cuenta\_bancaria, salario y estado

Si dos tuplas tienen el mismo nombre, nada (dos trabajadores pueden tener el mismo nombre, pero DNI distinto, telefono distinto…).

Si dos tuplas tienen el mismo teléfono, nada (dos trabajadores pueden tener el mismo teléfono de contacto, por ejemplo si es un telefono fijo…).

Si dos tuplas tienen la misma dirección, nada (dos trabajadores pueden tener la misma dirección si viven juntos por ejemplo).

Si dos tuplas tienen la misma cuenta\_bancaria, nada (una cuenta puede poseer varios titulares).

Si dos tuplas tienen el mismo salario, nada (una cuenta puede poseer varios titulares).

Si dos tuplas tienen el mismo estado, nada.

Se encuentra en FNBC ya que tanto DNI como correo y num\_seguridadsocial son superclaves.

## Normalización de la relación TrabajadorVendeVehiculo

TrabajadorVendeVehiculo(Matricula,fecha)

Dependencias funcionales no triviales: Matricula → fecha

Clave candidata: Matrícula

Se encuentra en FNBC ya que tanto DNI como Matrícula son superclaves.

# Subsistema Proveedores

## Normalización de la relación Proveedores

Proveedor(CIF, nombre\_empresa, telefono, direccion, correo, cuenta\_bancaria)

Dependencias funcionales no triviales

F = {

CIF → nombre\_empresa | telefono | direccion | correo | cuenta\_bancaria

}

Si dos tuplas tienen el mismo CIF, tienen que tener el mismo nombre, correo, direccion, cuenta\_bancaria.

Si dos tuplas tienen el mismo nombre\_empresa, nada (dos proveedores pueden tener el mismo nombre, pero CIF distinto, telefono distinto…).

Si dos tuplas tienen el mismo teléfono, nada (dos proveedores pueden tener el mismo teléfono de contacto, si son una empresa, si es un teléfono fijo…).

Si dos tuplas tienen la misma dirección, nada (dos proveedores pueden tener la misma dirección).

Si dos tuplas tienen el mismo correo, nada (dos proveedores pueden tener el mismo correo, correo de empresa…)

Si dos tuplas tienen la misma cuenta\_bancaria, nada (una cuenta puede poseer varios titulares).

Cualquier combinación de las anteriores no determina la misma tupla.

La única clave candidata es CIF, que es clave primaria. Como verifica la unicidad, CIF es superclave y la tabla está en FNBC.

## Normalización de la relación ProveedorSuministraVehículo

ProveedorSuministraVehiculo(Matricula, CIF, fecha, total-pagado)

Dependencias funcionales no triviales:

F = {

Matricula → CIF| Fecha | Total-pagado

}

Como un coche solo se puede suministrado una vez, si aparece la misma matricula debe ser la misma tupla.

Un proveedor puede suministrar varios coches el mismo día y al mismo precio.

La única clave candidata es Matricula, que es clave primaria. Como verifica la unicidad, Matricula es superclave y la tabla está en FNBC.

# 

# Subsistema Almacén

## Normalización de la relación Vehículos

Vehículos (Matricula, Marca, Modelo, CaracteristicasTecnicas, Kilometraje, AñoFabricacion, FechaEntrada)

A continuación, se proporciona el **listado de dependencias funcionales no triviales**:

Sea F el conjunto de dependencias funcionales dado por:

F = {Matricula → Marca, Matricula → Modelo, Matricula → CaracteristicasTecnicas,

Matricula → Kilometraje, Matricula → AñoFabricacion, Matricula → FechaEntrada, {Marca, Modelo} → CaracteristicasTecnicas}

La tabla Vehículos está en **primera forma normal (1FN)** dado que los dominios de los distintos atributos son atómicos.

**Normalización basada en dependencias: Segunda forma normal (2FN)**

Calculamos todas las Claves Candidatas (CC) del Esquema Vehículos (Matricula, Marca, Modelo, CarcteristicasTecnicas, Kilometraje, AñoFabricacion, FechaEntrada) con las dependencias funcionales no triviales dadas anteriormente:

| **Atributos No Derecha** | **Atributos Ambos** | **Atributos Solo Derecha** |
| --- | --- | --- |
| Matricula | Marca, Modelo | CaracteristicasTecnicas, Kilometraje, AñoFabricacion, FechaEntrada |

Así pues, el atributo Matricula estará en todas las Claves Candidatas de la tabla así como los atributos CaracteristicasTecnicas, Kilometraje, AñoFabricacion y FechaEntrada no estarán en ninguna Clave Candidata. Veamos que el cierre de Matricula vía las dependencias funcionales no triviales descritas anteriormente coincide con toda la relación:

{Matricula}F+ = {Matricula, Marca, Modelo, Kilometraje, AñoFabricacion, FechaEntrada, CaracteristicasTecnicas}

Luego Matricula es una Clave Candidata de la tabla Vehículos y además la única Clave Candidata de la misma.

Para ver si la relación está en segunda forma normal tenemos que ver que todos sus atributos no primos dependen de forma completa de las claves candidatas.

Los atributos no primos de la relación son: Marca, Modelo, CaracteristicasTecnicas, Kilometraje, AñoFabricacion y FechaEntrada.

Sin embargo, como todas las Claves Candidatas de esta relación son singleton (es decir, están integradas por un único atributo), la relación está en 2FN (puesto que todas las dependencias funcionales son completas).

**Normalización basada en dependencias: Tercera forma normal (3FN)**

Claramente la relación Vehículos no está en Tercera forma normal dado que tenemos la relación {Marca, Modelo} → CaracteristicasTecnicas en las cual {Marca, Modelo} no es superclave y CaracteristicasTecnicas no es atributo primo.

Es decir, la relación Vehículos (Matricula, Marca, Modelo, CaracteristicasTecnicas, Kilometraje, AñoFabricacion, FechaEntrada) presenta la siguiente dependencia funcional transitiva problemática:

Matricula → {Marca, Modelo}

{Marca, Modelo} → CaracteristicasTecnicas.

A continuación se detalla el proceso de normalización:

Empezaremos aplicando el *Teorema de Heath* a la dependencia {Marca, Modelo} → CaracteristicasTecnicas:

* Vehículos (Matricula, Marca, Modelo, Kilometraje, AñoFabricacion, FechaEntrada)
  + Primary Key: Matricula, dependencia “directa” (no transitiva). Tenemos que está relación está en Tercera Forma Normal dado que para toda dependencia no trivial ⍺ → β, se tiene que ⍺ es una superclave o β contiene solamente atributos primos. En efecto, las dependencias funcionales no triviales de esta relación son Matricula → Marca, Matricula → Modelo, Matricula → CaracteristicasTecnicas,

Matricula → Kilometraje, Matricula → AñoFabricacion y

Matricula → FechaEntrada y Matricula es superclave al ser una clave candidata.

* Caracteristicas (Marca, Modelo, CaracteristicasTecnicas)
  + Primary Key: (Marca, Modelo), dependencia “directa” (no transitiva). Tenemos que está relación está en Tercera Forma Normal dado que para toda dependencia no trivial ⍺ → β, se tiene que ⍺ es una superclave o β contiene solamente atributos primos. En efecto, la única dependencia funcional no trivial de esta relación es {Marca, Modelo} → CaracteristicasTecnicas y {Marca, Modelo} es superclave al ser una clave candidata.

**Forma Normal de Boyce-Codd**

La relación Vehículos está en Forma normal de Boyce-Codd (FNBC) dado que para toda dependencia no trivial ⍺ → β, se cumple que ⍺ es una superclave. En efecto, las dependencias funcionales no triviales de esta relación son Matricula → Marca, Matricula → Modelo, Matricula → CaracteristicasTecnicas, Matricula → Kilometraje, Matricula → AñoFabricacion y Matricula → FechaEntrada y Matricula es superclave al ser una clave candidata.

Asimismo, la relación Caracteristicas está en Forma normal de Boyce-Codd (FNBC) dado que para toda dependencia no trivial ⍺ → β, se cumple que ⍺ es una superclave. En efecto, la única dependencia funcional no trivial de esta relación es {Marca, Modelo} → CaracteristicasTecnicas y {Marca, Modelo} es superclave al ser una clave candidata.

## Normalización de la relación Disponibles

La relación Disponibles (Matricula) está ya en forma normal de Boyce-Codd (FNBC) dado que su esquema está constituido de un único atributo o singleton (con su dominio asociado) que además constituye la única clave primaria de la relación. Es decir, la relación ya está normalizada dado que no hay ninguna dependencia funcional no trivial y el dominio del atributo Matricula es atómico.

## Normalización de la relación Vendidos

Vendidos (Matricula, FechaSalida, Precio)

A continuación, se proporciona el **listado de dependencias funcionales no triviales**:

Sea F el conjunto de dependencias funcionales dado por:

F = {Matricula → FechaSalida, Matricula → Precio}

La tabla Vendidos está en **primera forma normal (1FN)** dado que los dominios de los distintos atributos son atómicos.

**Normalización basada en dependencias: Segunda forma normal (2FN)**

Calculamos todas las Claves Candidatas (CC) del Esquema Vendidos (Matricula, FechaSalida, Precio) con las dependencias funcionales no triviales dadas anteriormente:

| **Atributos No Derecha** | **Atributos Ambos** | **Atributos Solo Derecha** |
| --- | --- | --- |
| Matricula |  | FechaSalida, Precio |

Así pues, el atributo Matricula estará en todas las Claves Candidatas de la tabla así como los atributos FechaSalida y Precio no estarán en ninguna Clave Candidata. Veamos que el cierre de Matricula vía las dependencias funcionales no triviales descritas anteriormente coincide con toda la relación:

{Matricula}F+ = {Matricula, FechaSalida, Precio}

Luego Matricula es una Clave Candidata de la tabla Vendidos y además la única Clave Candidata de la misma.

Para ver si la relación está en segunda forma normal tenemos que ver que todos sus atributos no primos dependen de forma completa de las claves candidatas.

Los atributos no primos de la relación son: FechaSalida y Precio.

Sin embargo, como todas las Claves Candidatas de esta relación son singleton (es decir, están integradas por un único atributo), la relación está en 2FN (puesto que todas las dependencias funcionales son completas).

**Normalización basada en dependencias: Tercera forma normal (3FN)**

La relación Vendidos está en tercera forma normal (3FN) dado que para toda dependencia funcional no trivial ⍺ → β se cumple que ⍺ es una superclave o β contiene solamente atributos primos. En efecto, en la relación Vendidos tenemos las dependencias funcionales no triviales Matricula → FechaSalida y Matricula → Precio y Matricula es superclave al ser Clave Candidata.

**Forma Normal de Boyce-Codd**

La relación Vendidos está en Forma normal de Boyce-Codd (FNBC) dado que para toda dependencia no trivial ⍺ → β, se cumple que ⍺ es una superclave. En efecto, en la relación Vendidos tenemos las dependencias funcionales no triviales Matricula → FechaSalida y Matricula → Precio y Matricula es superclave al ser Clave Candidata.

## Normalización de la relación Desguazados

Desguazados (Matricula, FechaSalida, Precio)

A continuación, se proporciona el **listado de dependencias funcionales no triviales**:

Sea F el conjunto de dependencias funcionales dado por:

F = {Matricula → FechaSalida, Matricula → Precio}

La tabla Desguazados está en **primera forma normal (1FN)** dado que los dominios de los distintos atributos son atómicos.

**Normalización basada en dependencias: Segunda forma normal (2FN)**

Calculamos todas las Claves Candidatas (CC) del Esquema Desguazados (Matricula, FechaSalida, Precio) con las dependencias funcionales no triviales dadas anteriormente:

| **Atributos No Derecha** | **Atributos Ambos** | **Atributos Solo Derecha** |
| --- | --- | --- |
| Matricula |  | FechaSalida, Precio |

Así pues, el atributo Matricula estará en todas las Claves Candidatas de la tabla así como los atributos FechaSalida y Precio no estarán en ninguna Clave Candidata. Veamos que el cierre de Matricula vía las dependencias funcionales no triviales descritas anteriormente coincide con toda la relación:

{Matricula}F+ = {Matricula, FechaSalida, Precio}

Luego Matricula es una Clave Candidata de la tabla Desguazados y además la única Clave Candidata de la misma.

Para ver si la relación está en segunda forma normal tenemos que ver que todos sus atributos no primos dependen de forma completa de las claves candidatas.

Los atributos no primos de la relación son: FechaSalida y Precio.

Sin embargo, como todas las Claves Candidatas de esta relación son singleton (es decir, están integradas por un único atributo), la relación está en 2FN (puesto que todas las dependencias funcionales son completas).

**Normalización basada en dependencias: Tercera forma normal (3FN)**

La relación Desguazados está en tercera forma normal (3FN) dado que para toda dependencia funcional no trivial ⍺ → β se cumple que ⍺ es una superclave o β contiene solamente atributos primos. En efecto, en la relación Desguazados tenemos las dependencias funcionales no triviales Matricula → FechaSalida y Matricula → Precio y Matricula es superclave al ser Clave Candidata.

**Forma Normal de Boyce-Codd**

La relación Desguazados está en Forma normal de Boyce-Codd (FNBC) dado que para toda dependencia no trivial ⍺ → β, se cumple que ⍺ es una superclave. En efecto, en la relación Desguazados tenemos las dependencias funcionales no triviales

Matricula → FechaSalida y Matricula → Precio y Matricula es superclave al ser Clave Candidata.

# Subsistema Contabilidad

Mi subsistema tiene dos tablas, una que simplemente es un valor denominado Coste Fijo, esta se usará para calcular los costes totales y los ingresos totales. La otra es el registro de todos pagos realizados a los trabajadores.

### Normalización de la relación Contabilidad

Contabilidad(#Coste\_Fijo)

La única clave y por tanto clave candidata y clave primaria es #Coste\_Fijo. Como verifica la unicidad, es superclave y la tabla está en FNBC.

### Normalización de la relación Registro Salarios

Registro Salarios(#DNI, #Fecha\_Pago, Salario)

En el caso de la tabla de Registro Salarios, #Fecha\_Pago es principal junto con #DNI manteniendo una relación débil con Trabajadores donde #DNI es la clave externa.

Es una FNBC porque nos aseguramos de que para cada combinación solo es posible una entrada.

Si dos tuplas tienen el mismo DNI, tienen que tener distinta Fecha de Pago para entender que son pagos de meses diferentes.

Si dos tuplas tienen la misma Fecha de Pago, tienen que tener distinto DNI, porque sino se le habría pagado dos veces al mismo trabajador.

Si dos tuplas tienen el mismo salario nada, dos trabajadores pueden tener el mismo salario.

Sentencias SQL de creación de tablas obtenidas y de inserción de tuplas

A continuación, se presenta un enlace a un archivo con las sentencias de creación de las relaciones del sistema “NOMBRE”, con todas las restricciones previamente especificadas:

[CREACIÓNTABLAS.sql](https://drive.google.com/file/d/1bRgWG0mVBRyc5rTAGShFJ8ZAeLDm9QPm/view?usp=share_link)

Por último, se muestran las sentencias SQL de inserción de 10 de tuplas en las tablas del sistema de información “NOMBRE”:

[INSERCIONTUPLAS.sql](https://drive.google.com/file/d/138hhcVaajuzxA-X9BlcxUQpdwRhBiwaW/view?usp=share_link)

Correcciones de la práctica 2.

#### Subsistema Clientes:

Correcciones en el esquema externo de nivel 1.

Correcciones en el diagrama de entidad relación.

Correcciones en el paso a tablas.

#### Subsistema Trabajadores:

Correcciones en la creación de tablas

Correcciones en el esquema externo de añadir trabajadores

#### Subsistema Proveedores:

Correcciones en el esquema externo de pedir vehículo. Ahora se llama suministrar vehículo y en la relación se almacena la fecha de suministro y el precio.

#### Subsistema Almacén:

* Se han modificado todos los esquemas externos de nivel 1.
* Se ha modificado el esquema externo de nivel 0 del proceso almacén.
* Se ha modificado el diagrama E/R.
* Se ha modificado consecuentemente el paso a tablas.
* Se ha modificado el proceso de normalización.
* Se han modificado las sentencias de creación de tablas y de inserción de tuplas en las tablas creadas, como consecuencia de las modificaciones realizadas previamente.
* Se ha modificado la restricción semántica 4.2.

#### Subsistema Contabilidad:

* Correcciones en los esquemas externos de nivel 1 después de las modificaciones sugeridas por el profesor.
* Cambiado el diseño de cómo se gestiona el registro de salarios. Decidí que era más sencillo de implementar y tenía más sentido que se guardaran en el Registro de Salarios tanto el DNI, como la FechaPago como el salario. Primero porque así no tengo que gestionar entidades debiles y porque esto me permite tener un histórico fiel (puedes modificar el salario al trabajador y solo se actualizan las nuevas entradas).
* En la RF5.5 he empezado a usar TrabajadorVendeVehiculo en lugar de Trabajadores.
* También he empezado a usar ProveedorSuministraVehiculo en lugar de Proveedores.

# 

# Sentencias SQL Disparadores

A continuación, se presenta un enlace a un archivo con las sentencias de creación de los disparadores del sistema “NOMBRE”:

[DISPARADORES.sql](https://drive.google.com/file/d/13CsuC2obqO61DQAT5Bo3HMKNcS4BKvzf/view?usp=share_link)