Grupo 3

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Mercadona

Mercadona, empresa de capital familiar, es una de las principales compañías de supermercados físicos y online en España que tiene por objetivo asumir la responsabilidad de prescribir al cliente la mejor opción para satisfacer sus necesidades de alimentación, limpieza del hogar, higiene personal y cuidado de mascotas.
Fue fundada en 1977 por el Grupo Cárnicas Roig, actualmente dispone de 1.624 tiendas en toda España y 16 en Portugal, y una plantilla de 90.000 personas.

- **Misión.** La misión es descrita por la propia empresa como:
  "Prescribir al consumidor final productos / soluciones que cubran sus necesidades de comer, beber, cuidado personal, cuidado del hogar y cuidado de animales asegurando siempre Seguridad Alimentaria, Máxima Calidad, Máximo Servicio, Mínimo Presupuesto y Mínimo Tiempo".

- **Visión.** La visión es descrita por la propia empresa como:
  “Conseguir una Cadena Agroalimentaria Sostenible, que la gente quiera que exista y sienta orgullo de ella, liderada por Mercadona y teniendo a ‘El Jefe’ como faro”.

1. Logística (Ángel Amadeo González Ruiz)

La función de la logística es la planificación y la gestión del flujo de materiales de la manera más eficaz entre nuestros proveedores y nuestros clientes finales, incluyendo la creación e implementación de sistemas de control y mejora, aclarada esta idea principal a destacar de la compañía Mercadona, es que cuenta con 1372 tiendas en España que se abastecen a través de 13 bloques logísticos y de una red logística que, además de la máxima eficiencia, persigue transportar más con menos recursos y ser cada vez más invisible para el entorno.

Respecto a la distribución, Mercadona lleva a cabo una tercera parte de las operaciones de abastecimiento de sus supermercados en horario nocturno, entre las 10 de la noche y las 6 de la mañana, mientras que el resto de operaciones de suministro a las tiendas se lleva a cabo durante el día.

La logística de abastecimiento de productos de Mercadona pasa porque todos sus proveedores descargan el género en los centros de distribución de la compañía, con lo que ningún suministrador entrega directamente en tienda.

Esta organización permitió a Mercadona un ahorro de 20 millones de euros en costes logísticos gracias al acercamiento de las instalaciones de los proveedores a los bloques logísticos, así como a la optimización de la capacidad y del nivel de ocupación de los camiones

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## Requisitos Funcionales

RF1.1	**Mostrar pedidos en curso**
Entrada	**Agente externo empleado de logística: Acción: solicitar listado. Requisito de datos de entrada: ninguno**
BD	**Requisito de datos de lectura: RDR1.1**
Salida	**Agente externo empleado de logística: Acción: confirmación resultado. Requisito de datos de salida: RDS1.1**
RDE1.1	
RDR1.1	**Datos de los pedidos en curso**
RDW1.1	
RDS1.1	**Listado de todas los pedidos, su precio y fecha que toma de RDR1.1**

RF1.2	**Consultar pedido**
Entrada	**Agente externo: empleado de logística. Acción: solicitar datos Requisito de datos de entrada: RDE1.2**
BD	**Requisito de datos de escritura RDW1.2**
Salida	**Agente externo:empleado de logística. Acción: listado con la información**
RDE1.2	**Datos de entrada de código del pedido Descripción: Cadena de caracteres(50) Materiales: Cadena de caracteres(50)**
RDR1.2	
RDW1.2	
RDS1.2	

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF1.3	**Cancelar pedido**
Entrada	**Agente externo: empleado de logística. Acción: solicitar cancelación de pedido** **Requisito de datos de entrada: RDE1.3**
BD	**Requisitos de datos de lectura RDR1.3**
Salida	**Agente externo: empleado de logística. Acción: confirmación de la cancelación del pedido: RDS1.3**
RDE1.3	**Datos de entrada sobre el pedido a retirar** **Número identificativo: cadena de caracteres(25)**
RDR1.3	**Datos sobre el pedido a cancelar**
RDW1.3	
RDS1.3	**Confirmación sobre la correcta devolución del pedido**

RF1.4	**Registrar pedido**
Entrada	**Agente externo: empleado de logística. Acción: registrar un nuevo pedido realizado** **Requisito de datos de entrada: ninguno**
BD	**Requisito de datos de lectura RDR1.4**
Salida	**Agente externo:empleado de logística. Acción: confirmación resultado** **Requisito de datos de salida: RDS3**
RDE1.4	**El pedido como tal y su destinatario**
RDR1.4	**Registro del nuevo pedido**
RDW1.4	**Registrar el pedido realizado**
RDS1.4	**Confirmación de que se ha realizado correctamente el registro**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF1.5	**Modificar pedido**
Entrada	**Agente externo: empleado de logística. Acción: modificar un pedido en curso** **Requisito de datos de entrada: RDE1.5**
BD	**Requisito de datos de lectura RDR1.5**
Salida	**Agente externo empleado de logística: Acción: confirmación de resultado.** **Requisito de datos de salida: RDS3**
RDE1.5	**Código de pedido asociado** **Número de identificación: cadena de caracteres(20).**
RDR1.5	
RDW1.5	
RDS1.5	**Confirmación de la agregación de la nueva modificación**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## Requisitos Semánticos

RS1.1	**Deberá existir al menos un pedido en curso**
RF	**RF1.1**
RD	**RDE1.1, RDR1.1**
Descripción	**“Si no se encuentra ningún pedido en curso en sistema devuelve un error”.**

RS1.2	**El código debe existir**
RF	**RF1.2**
RD	**RDE1.2**
Descripción	**“Si el código no existe devolverá un código de error”.**

RS1.3	**La devolución deberá ser en un plazo inferior a 15 días**
RF	**RF1.3**
RD	**RDE1.3, RDR1.3**
Descripción	**“Si el pedido ya se recibió el pedido no se podrá cancelar”.**

RS1.4	**Deberá existir el producto del que se quiere hacer el pedido**
RF	**RF1.4**
RD	**RDE1.4**
Descripción	**“Si no existe el producto se enviará un aviso de fallo en el registro de este”.**

RS1.5	**No podremos modificar un pedido que no esté en curso**
RF	**RF1.5**
RD	**RDE1.5, RDR1.5**
Descripción	**“Si queremos modificar datos sobre un pedido que ya ha finalizado no se podrá y devolverá un código de error**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

2. Recursos Humanos (Carlos García Segura)

La función de los recursos humanos es alinear a los empleados con la estrategia de la organización mediante el reclutamiento, formación, desarrollo y gestión del personal de la empresa. Además de encargarse de las necesidades de los empleados, asegurandose de la igualdad de oportunidades entre estos combatiendo cualquier tipo de discriminación y estableciendo un ambiente de trabajo comodo y motivador mediante el establecimiento de reglas y politicas justas para los empleados asegurandose de no disminuir el rendimiento de estos.

Mercadona, dice en su web, “basa su política de Recursos Humanos en la verdad universal de que “para poder recibir, primero hay que dar”. Para conseguirlo, la compañía mantiene una política de Recursos Humanos que fomenta valores irrenunciables y esenciales para las personas:

- **Estabilidad:** Empleo estable y de calidad.
- **Igualdad:** A misma responsabilidad, mismo sueldo.
- **Formación y Promoción:** 106 millones de euros invertidos en formación y 849 personas promocionadas al año.
- **Conciliación:** En 2019, 2.284 personas optaron por alargar su permiso de nacimiento 30 días y 15.899 personas disfrutaron de jornada reducida.
- **Retribución:** Reparto de los beneficios obtenidos a lo largo del año. La compañía comparte el 25% del beneficio anual entre la plantilla.”

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## Requisitos Funcionales

RF2.1	**Solicitar Información de Nóminas**
Entrada	**Agente externo: Empleado de Recursos Humanos. Acción: Solicitar nómina del trabajador.** **Requisito de datos de entrada y de salida.**
BD	**Requisitos de datos de lectura**
Salida	**Agente externo: Empleado de Recursos Humanos. Acción: confirmación de resultado** **Requisito de datos de entrada.**
RDE2.1	**Documento de identificación del empleado: cadena de caracteres(20)**
RDR2.1	**Datos de las nóminas del empleado**
RDW2.1	
RDS2.1	**Nómina a pagar**

RF2.2	**Control de Ausencias**
Entrada	**Agente externo: Empleado de Recursos humanos. Acción: Apuntar falta de un trabajador.** **Requisito de datos de entrada y de salida**
BD	**Requisitos de datos de escritura y lectura**
Salida	**Agente externo: Empleado de Recursos Humanos. Acción: confirmación de resultado** **Requisito de datos de salida.**
RDE2.2	**Documento de Identificación del empleado: cadena de caracteres(20)**
RDR2.2	**Información sobre las faltas de un trabajador.**
RDW2.2	**Fecha de la falta: Date**
RDS2.2	**Mismos datos que en RDR2.2.**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF2.3	**Dar de Alta Empleado**
Entrada	**Agente externo: Empleado de Recursos Humanos. Acción: Dar de Alta nuevo empleado** **Requisito de datos de entrada y de salida**
BD	**Requisitos de datos de escritura**
Salida	**Agente externo: Empleado de Recursos Humanos. Acción: confirmación de resultado** **Requisito de datos de salida.**
RDE2.3	**Nombre y apellidos: cadena de caracteres(20)** **Documento de Identificación: cadena de caracteres(20)** **Fecha de Nacimiento: Date** **Teléfono: cadena de caracteres(9)**
RDR2.3	
RDW2.3	**Misma información que en RDE2.3**
RDS2.3	

RF2.4	**Dar de Baja Empleado**
Entrada	**Agente externo: Empleado de Recursos Humanos. Acción: Dar de Alta nuevo empleado** **Requisito de datos de entrada y de salida**
BD	
Salida	**Agente externo: Empleado de Recursos Humanos. Acción: confirmación de resultado** **Requisito de datos de salida.**
RDE2.4	**Documento de Identificación: cadena de caracteres(20)**
RDR2.4	
RDW2.4	**Confirmar baja**
RDS2.4	

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF2.5	**Confirmar Asistencia**
Entrada	**Agente externo: Empleado de Recursos Humanos. Acción: Apuntar su asistencia al trabajo.** **Requisito de datos de entrada y de salida**
BD	**Requisitos de datos de escritura y lectura**
Salida	**Agente externo: Empleado. Acción: confirmación de resultado** **Requisito de datos de salida.**
RDE2.5	**Documento de identificación del empleado: cadena de caracteres(20)**
RDR2.5	**Faltas y asistencias del empleado**
RDW2.5	**Confirmacion de asistencia**
RDS2.5	

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## Requisitos Semánticos

RS2.1	**El formato del documento de identificación debe ser el correcto**
RF	**RF2.1**
RD(s)	**RDE2.1**
Descripción	**El formato del documento de identificación no es correcto, no se introducen los datos en la base de datos, devolviendo un mensaje de error solicitando la corrección de los datos.**

RS2.2	**El formato del documento de identificación debe ser el correcto**
RF	**RF2.2**
RD(s)	**RDE2.2**
Descripción	**El formato del documento de identificación no es correcto, no se introducen los datos en la base de datos, devolviendo un mensaje de error solicitando la corrección de los datos.**

RS2.3	**El formato de los Datos de Entrada debe ser correcto**
RF	**RF2.3**
RD(s)	**RDE2.3**
Descripción	**Si el formato de los datos de entrada no es correcto, no se introducen los datos en la base de datos, devolviendo un mensaje de error solicitando la corrección de los datos.**

RS2.4	**El formato del documento de identificación debe ser el correcto**
RF	**RF2.4**
RD(s)	**RDE2.4**
Descripción	**El formato del documento de identificación no es correcto, no se introducen los datos en la base de datos, devolviendo un mensaje de error solicitando la corrección de los datos.**

RS2.5	**El formato del documento de identificación debe ser el correcto**
RF	**RF2.5**
RD(s)	**RDE2.5**
Descripción	**Si el formato del número de identificación no es correcto, no se realiza la consulta y se pide que se corrija.**

3. Financiero (Santiago Carbó García)

Las finanzas de una empresa la lleva a cabo el departamento financiero. Este se ocupa de las responsabilidades económicas de la empresa, así como realizar los pagos a los que está obligada dicha empresa y la gestión de las partidas de gastos e ingresos de la misma. La actividad de este departamento tiene una importancia crucial ya que el objetivo final de una empresa es la creación de valor, con el cual obtiene beneficios. También se encarga de la planificación y elaboración de presupuestos, pagos de nóminas, modelos de organización financiera…

Mercadona es una empresa que consigue, anualmente, un resultado de la explotación de 800 millones de euros y unos ingresos de 20 mil millones de euros. El encargado de gestionar la empresa a nivel financiero es Héctor Hernández.

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## Requisitos Funcionales y Secuenciales

RF3.1	**Mostrar pagos**
Entrada	**Agente externo: empleado de finanzas.** **Acción: solicitar listado.** **Requisitos de datos de entrada: ninguno.**
BD	**Requisito de datos de lectura RDR3.1**
Salida	**Agente externo: empleado de finanzas.** **Acción: confirmación resultado.** **Requisitos de datos de salida**
RDR3.1	**Datos del pago almacenado:** **Código: Cadena de caracteres (20)** **Fecha: Cadena de caracteres (20)** **Capital: Cadena de caracteres (30)** **Interés: Cadena de caracteres (20)** **Dividendo: Cadena de caracteres (30)**
RDS3.1	**Listado de registros, cada uno de ellos con los mismos datos de RDR3.1**

RS3.1	**Deberá existir al menos un pago**
RF	**RF3.1**
RD	**RDR3.1**
Descripción	**“Si no se encuentra ningún pago realizado en el sistema devuelve un error”.**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF3.2	**Registrar pago**
Entrada	**Agente externo: empleado de finanzas** **Acción: solicitar registro de nuevo pago** **Requisitos de entrada RDE3.2**
BD	**Requisito de datos de escritura RDW3.2**
Salida	**Agente externo: empleado de finanzas** **Acción: confirmación de resultado** **Requisitos de datos de salida:ninguno**
RDE3.2	**Datos de entrada del pago: los mismos datos que RDR3.1**
RDW3.2	**Datos almacenados del pago: los mismos datos que RDE3.1**

RS 3.2	**Los datos de entrada, así como el formato del pago deben ser correctos**
RF	**RF 3.2**
RD	**RDE 3.2**
Descripción	**“En el caso de que algún campo esté incompleto aparecerá un error pidiendo que se rellene dicho campo”**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF3.3	**Eliminar pago**
Entrada	**Agente externo: empleado de finanzas** **Acción: solicitar borrado de un pago** **Requisitos de entrada RDE3.3**
BD	**Requisito de datos de escritura RDW3.3**
Salida	**Agente externo: empleado de finanzas** **Acción: confirmación de resultado** **Requisitos de datos de salida:ninguno**
RDE3.3	**Datos de entrada del pago: Código: Cadena de caracteres (20)**
RDW3.3	**Datos almacenados del pago: los mismos datos que RDW3.2**

RS 3.3	**El código del pago indicado debe ser correcto**
RF	**RF 3.3**
RD	**RDE 3.3**
Descripción	**“En el caso de que el código del producto sea incorrecto se indica al usuario que lo corrija”**

RF3.4	**Mostrar partidas**
Entrada	**Agente externo: empleado de finanzas.** **Acción: solicitar listado.** **Requisitos de datos de entrada: ninguno.**
BD	**Requisito de datos de lectura RDR3.4**
Salida	**Agente externo: empleado de finanzas.** **Acción: confirmación resultado.** **Requisitos de datos de salida**
RDR3.4	**Datos del pago almacenado:** **Capítulo: Cadena de caracteres (20)** **Importe: Cadena de caracteres (20)** **Porcentaje: Cadena de caracteres (10)**
RDS3.4	**Listado de registros, cada uno de ellos con los mismos datos de RDR3.4**

RS3.4	**Deberá existir al menos una partida**
RF	**RF3.4**
RD	**RDR3.4**
Descripción	**“Si no se encuentra una partida aparecerá el oportuno error”.**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF3.5	**Registrar partida**
Entrada	**Agente externo: empleado de finanzas** **Acción: solicitar nuevo gasto** **Requisitos de entrada RDE3.5**
BD	**Requisito de datos de escritura RDW3.5**
Salida	**Agente externo: empleado de finanzas** **Acción: confirmación de resultado** **Requisitos de datos de salida:ninguno**
RDE3.5	**Datos de entrada del pago: los mismos datos que RDR3.4**
RDW3.5	**Datos almacenados del pago: los mismos datos que RDE3.4**

RS 3.5	**Los datos de entrada, así como el formato de la partida deben ser correctos**
RF	**RF 3.5**
RD	**RDE 3.5**
Descripción	**“En el caso de que algún campo esté incompleto aparecerá un error pidiendo que se rellene dicho campo”**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

4. Proveedores y almacén

Mercadona, junto a los 1.400 Proveedores Totaler con los que colabora, persigue la excelencia para construir un surtido eficaz que aporte diferenciación y proporcione productos con la máxima calidad y mínimo coste.

Además, apuesta por la especialización en frescos, con compras de productos locales y de cercanía para una mayor frescura.

De esta forma los proveedores comerciales y de servicios y pymes con las que indirectamente trabaja, asumen un papel relevante en la cadena de montaje de Mercadona, pues su implicación conjunta ha permitido seguir promoviendo y consolidando un proyecto de crecimiento compartido que genera valor en los entornos en los que está presente.

● Requisitos Funcionales

RF 4.1	**Eliminar producto**
Entrada	**Agente externo: Empleado almacen** **Acción: solicitar eliminar producto** **Requisito de datos de entrada: Identificador de producto. RDE 4.1**
BD	**Requisitos de datos de escritura RDW 4.1**
Salida	**Agente externo: Empleado almacen** **Acción: confirmación resultado** **Requisito de datos de salida: ninguno**
RDE 4.1	**Identificador de producto a eliminar** **ID: cadena de caracteres (10)**
RDR 4.1	
RDW 4.1	**Datos almacenados del producto.**
RDS 4.1	

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF 4.2	**Consultar Stock**
Entrada	**Agente externo: Empleado almacen** **Acción: solicitar disponibilidad de producto** **Requisito de datos de entrada: Identificador de producto.**
BD	**Requisitos de datos de lectura RDR 4.2**
Salida	**Agente externo: Empleado almacen** **Acción: confirmación de información** **Requisito de datos de salida RDS 4.2**
RDE 4.2	**Identificador de producto consulado** **ID: cadena de caracteres(20)**
RDR 4.2	**Disponibilidad del producto consultado**
RDW 4.2	
RDS 4.2	**Información sobre el producto que se consulta** **Datos de RDR 4.2**

RF 4.3	**Mostrar lista de proveedores**
Entrada	**Agente externo: Empleado almacen** **Acción: solicitar listado** **Requisito de datos de entrada: ninguno.**
BD	**Requisitos de datos de lectura**
Salida	**Agente externo: Empleado almacen** **Acción: confirmación resultado** **Requisito de datos de salida: RDS 4.3**
RDE 4.3	
RDR 4.3	**Datos de los proveedores registrados** **Los mismos datos que en RDW 4.5**
RDW 4.3	
RDS 4.3	**Listado de proveedores, cada uno de ellos con los mismos datos de RDR 4.3**

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF 4.4	**Registrar pedido a proveedor**
Entrada	**Agente externo: Empleado almacen** **Acción: solicitar registro de nuevo pedido** **Requisitos de entrada 4.4**
BD	**Requisito de datos de escritura**
Salida	**Agente externo: Empleado almacen** **Acción: confirmación de resultado** **Requisitos de datos de salida: ninguno**
RDE 4.4	**Datos de entrada de nuevo pedido** **Código de pedido: cadena de caracteres de (20)** **Proveedor involucrado: cadena de caracteres (30)** **Identificador productos: cadena de caracteres (10)** **Cantidad de producto: cadena de caracteres (3)** **Precio del pedido: cadena de caracteres (10)**
RDR 4.4	
RDW 4.4	**Datos del pedido registrado** **Mismo que RDE 4.4**
RDS 4.4	

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

RF 4.5	**Registrar proveedores**
Entrada	**Agente externo: Empleado almacen** **Acción: solicitar inserción de nuevo proveedor** **Requisitos de entrada RDE 4.5**
BD	**Requisito de datos de escritura**
Salida	**Agente externo: Empleado almacen** **Acción:confirmación de resultado** **Requisitos de datos de salida: ninguno**
RDE 4.5	**Datos de entrada de nuevo proveedor** **Nombre: cadena de caracteres de (20)** **Apellidos: cadena de caracteres (30)** **DNI: cadena de caracteres (10)** **Contacto: cadena de caracteres (25)**
RDR 4.5	
RDW 4.5	**Datos del nuevo proveedor** **Los mismo que RDE 4.5**
RDS 4.5	

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## Requisitos Semánticos

RS 4.1	**La ID del producto que se quiere eliminar debe ser válida**
RF	**RF 4.1**
RD(s)	**RDE 4.1**
Descripción	**Si el formato del número de identificación no es válido, no se procede con la eliminación del producto y se pide que se corrija.**

RS 4.2	**La ID del producto que se quiere eliminar debe ser válida**
RF	**RF 4.2**
RD(s)	**RDE 4.2**
Descripción	**Si el formato del número de identificación no es válido, no se realiza la consulta y se pide que se corrija.**

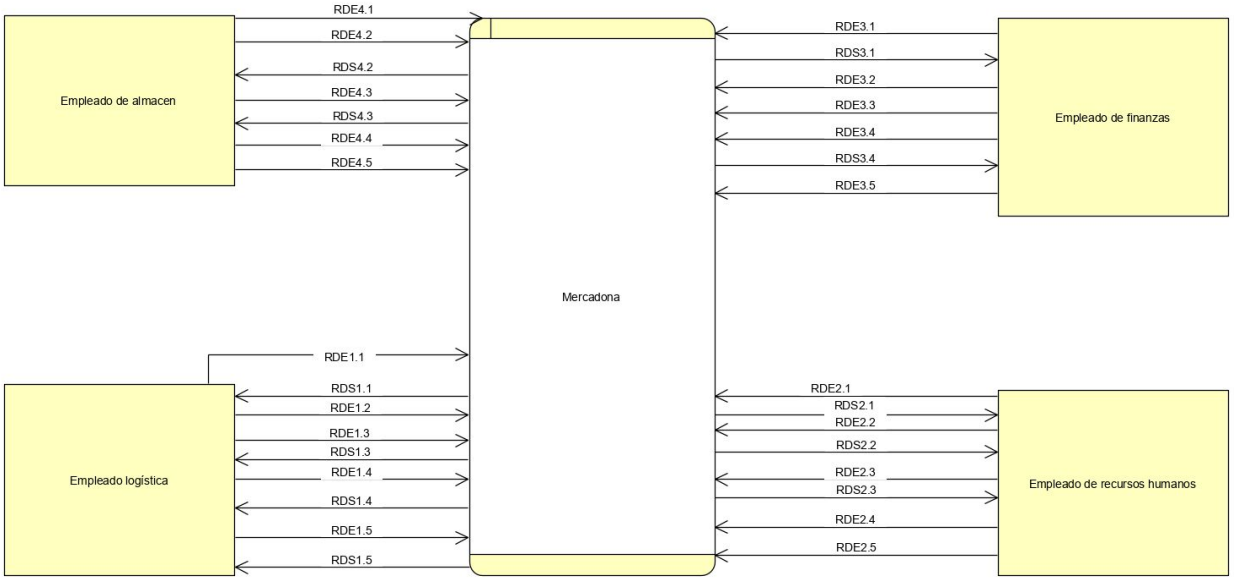
RS 4.3	**Debe de haber al menos un proveedor registrado**
RF	**RF 4.3**
RD(s)	**RDE 4.3, RDR 4.3**
Descripción	**Si no se encuentra ningún proveedor registrado, se devuelve un error.**

RS 4.4	**El formato del pedido debe ser correcto y todos los campos deben de estar correctamente definidos, los productos incluidos en el pedido deben de estar disponibles.**
RF	**RF 4.4**
RD(s)	**RDE 4.4**
Descripción	**Si alguno de los datos necesarios para el encargo del pedido está incompleto, se envía un mensaje al usuario para que complete los datos convenientemente.**

RS 4.5	**Los datos de registro deben de ser correctos y completos y el proveedor no puede estar ya registrado.**
RF	**RF 4.5**
RD(s)	**RDE 4.5**
Descripción	**Si el formato de los datos del proveedor es incorrecto o están incompletos, se le pide al usuario que los complete mediante un mensaje.**

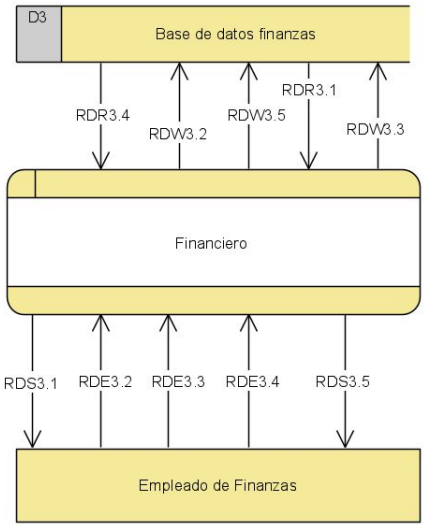
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

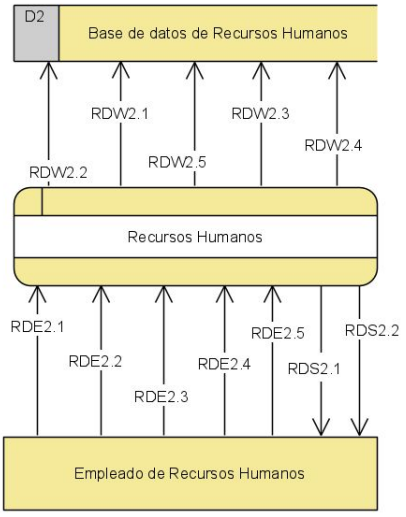
- **Caja Negra:**

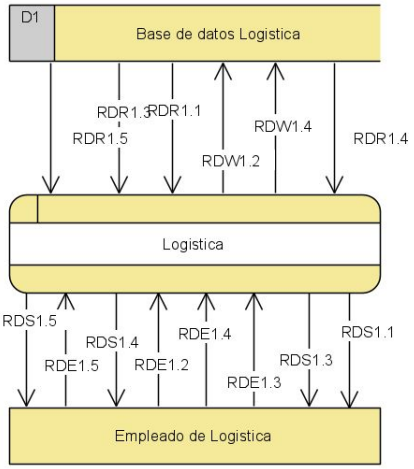


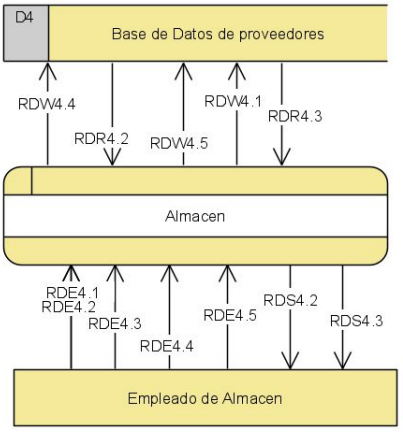
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Armazón:**





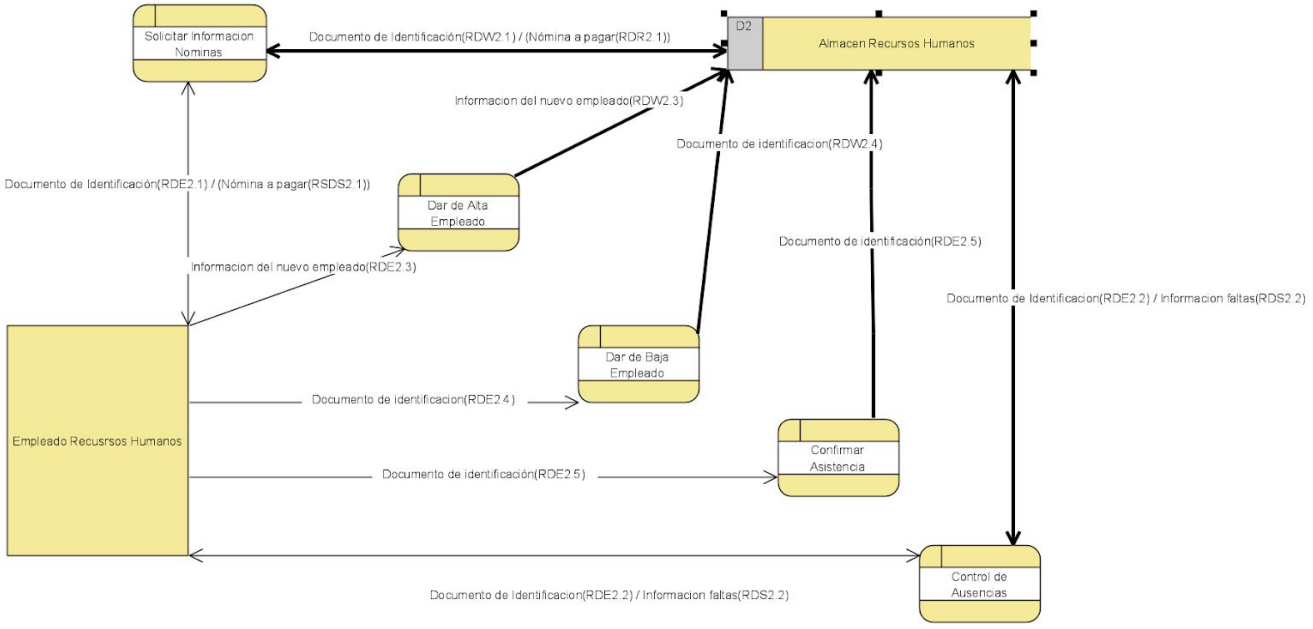




Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

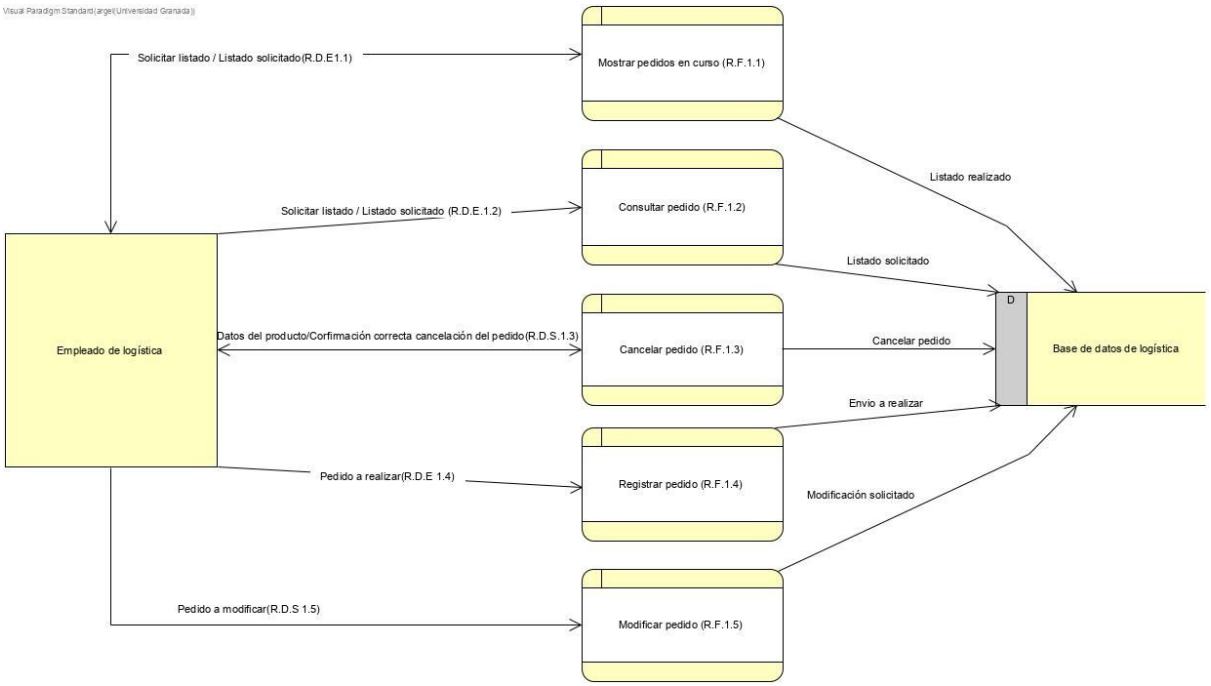
- **DFD1**

Carlos García Segura



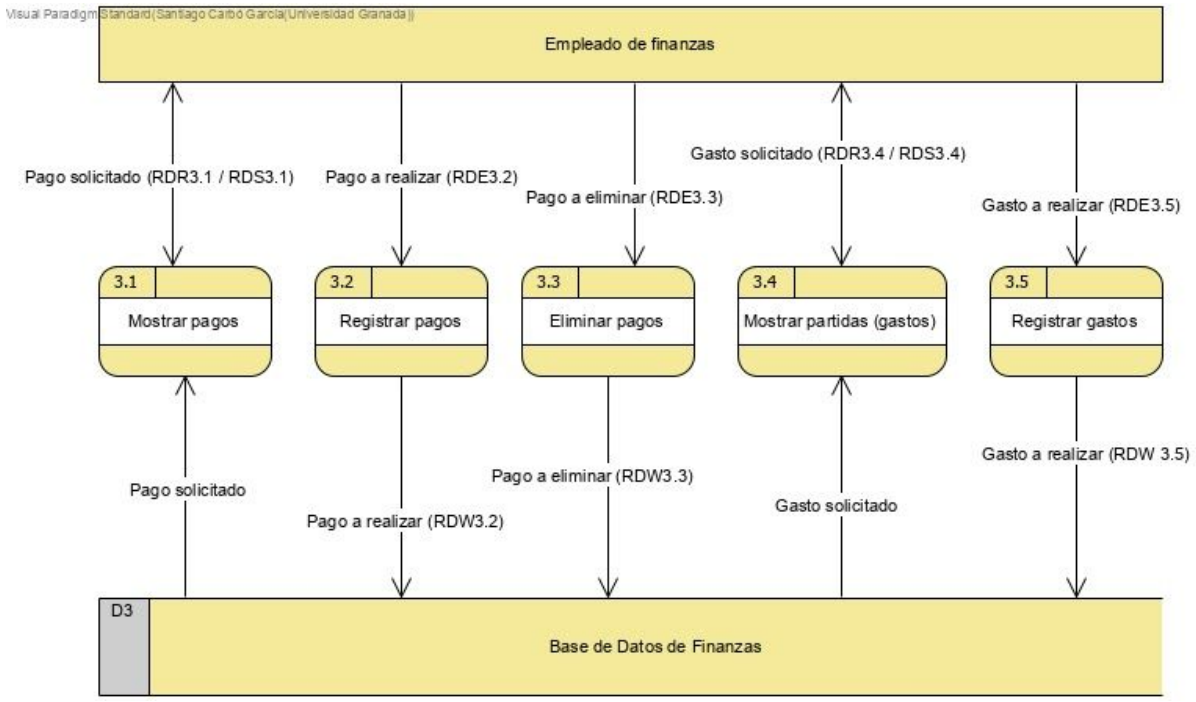
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Angel Amadeo Gonzalez Ruiz



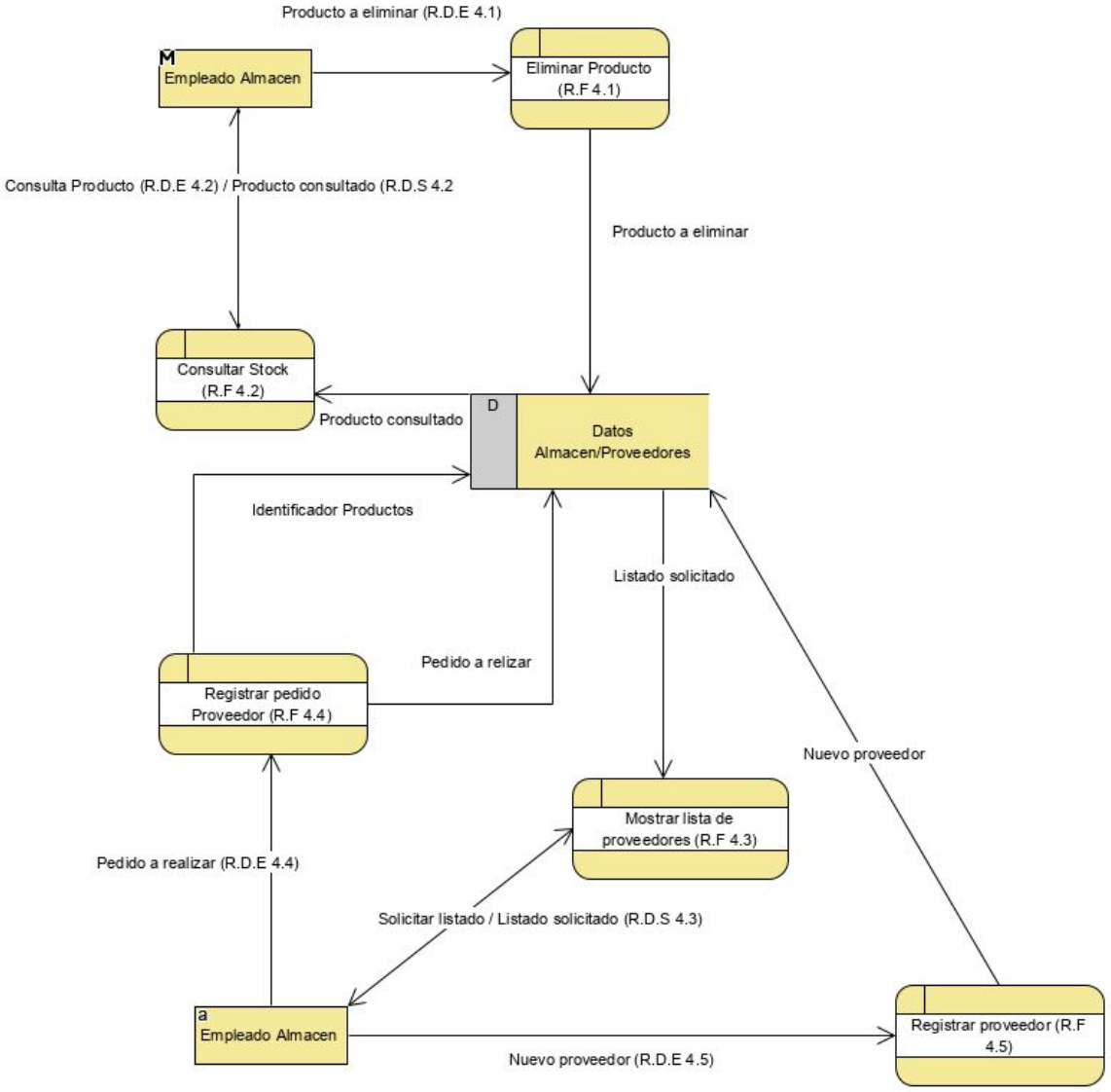
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Santiago Carbó García



Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

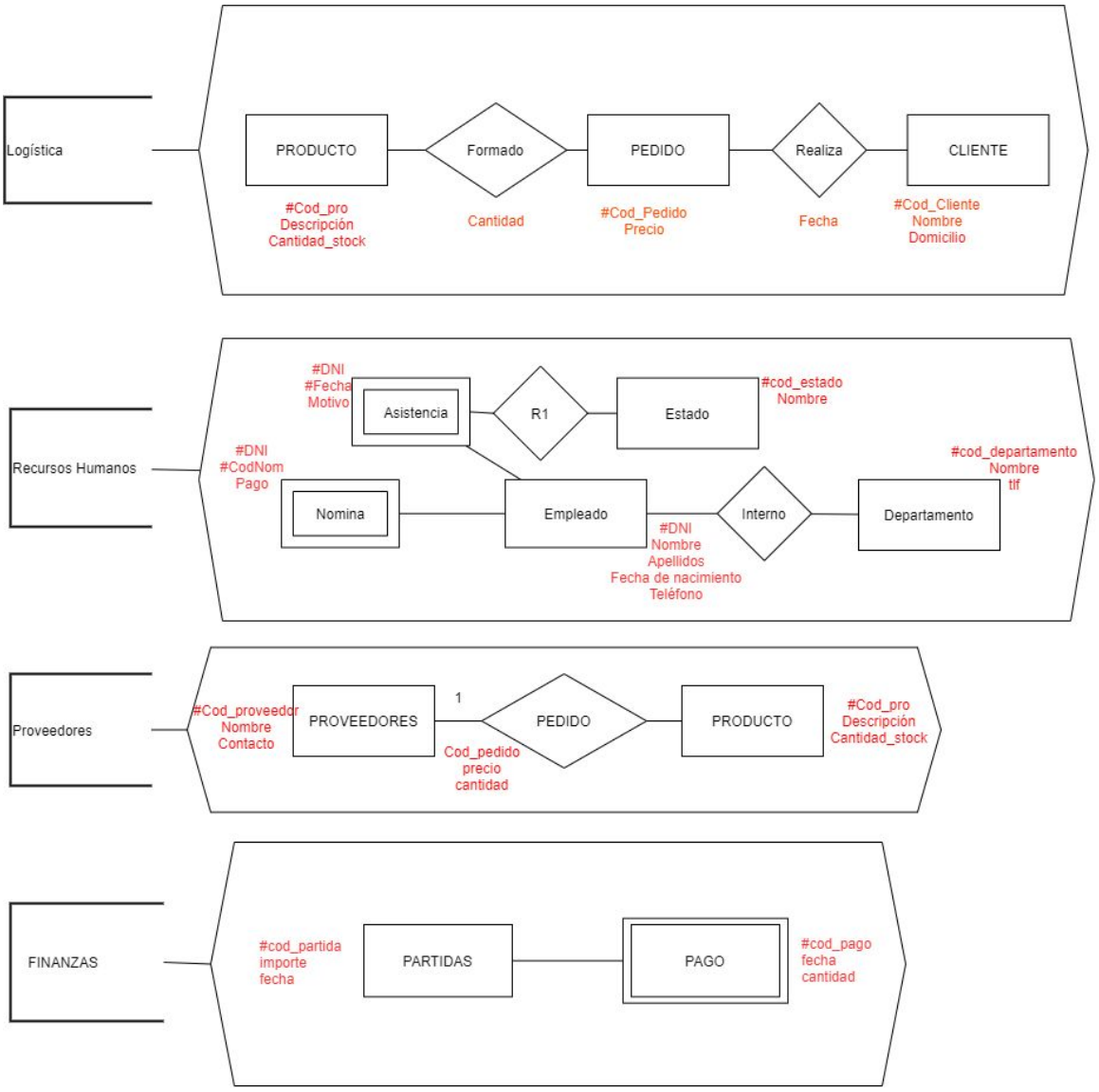
Manuel Jose Cano Rojo



Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

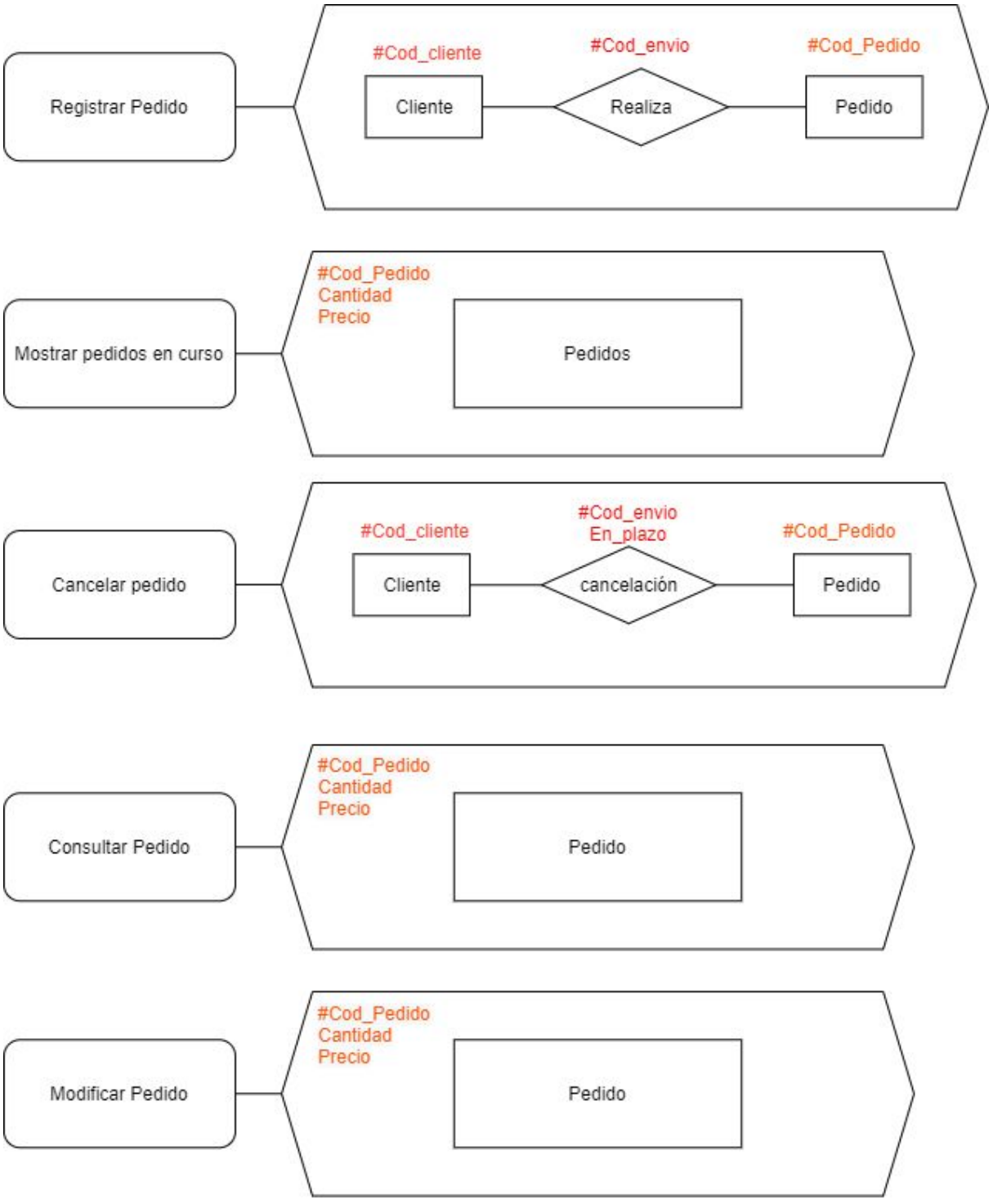
ESQUEMAS EXTERNOS

Armazón



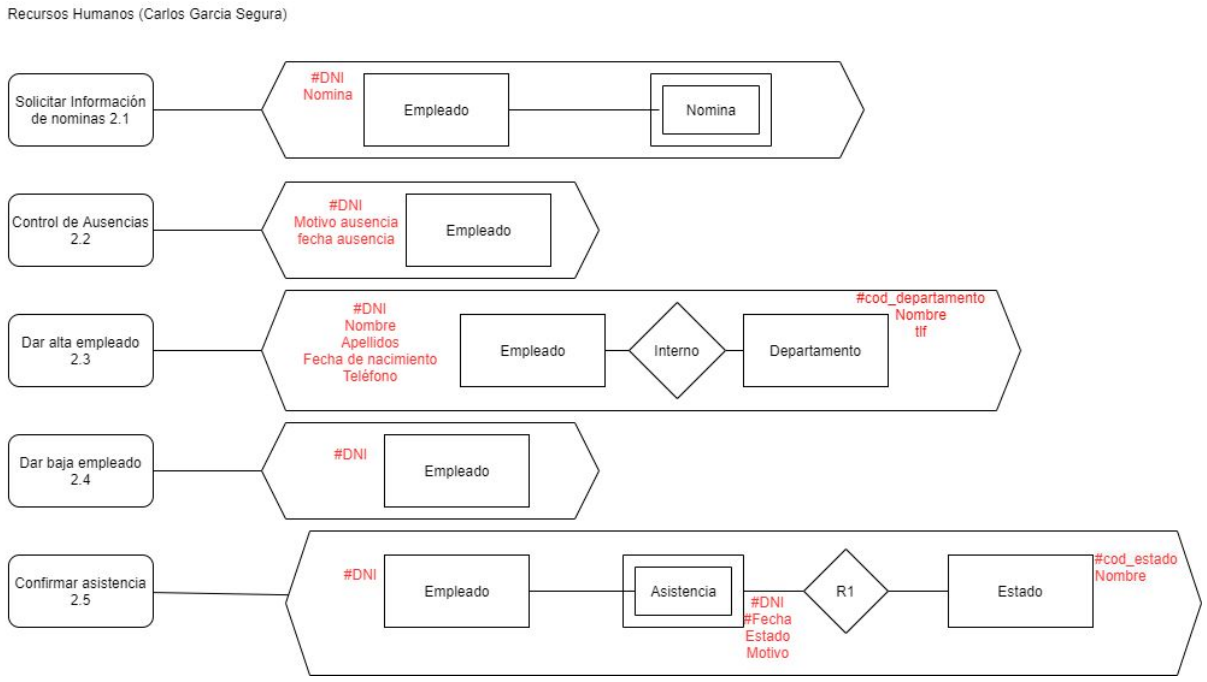
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Ángel Amadeo González Ruiz (Logistica)



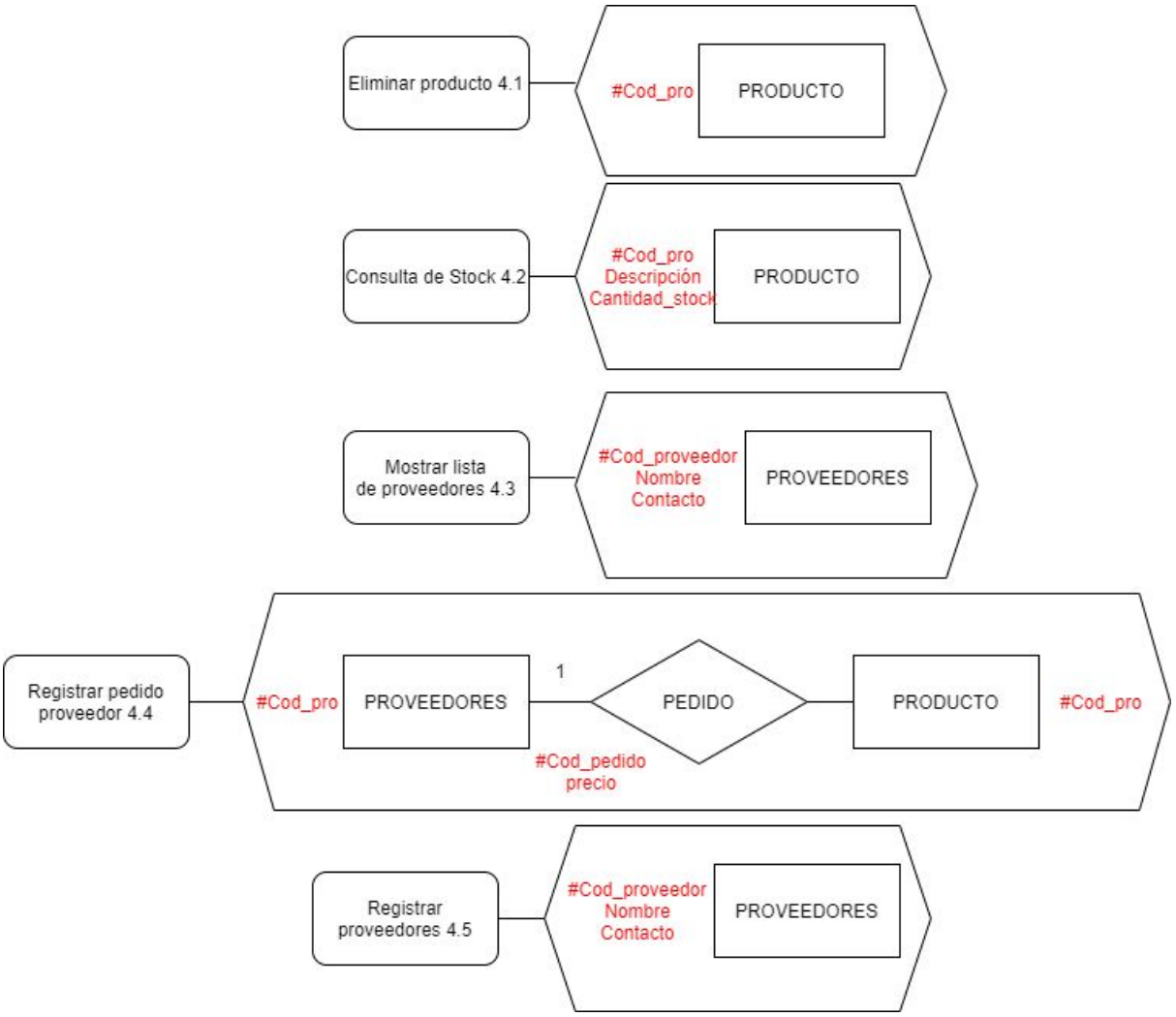
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Carlos García Segura (Recursos Humanos)



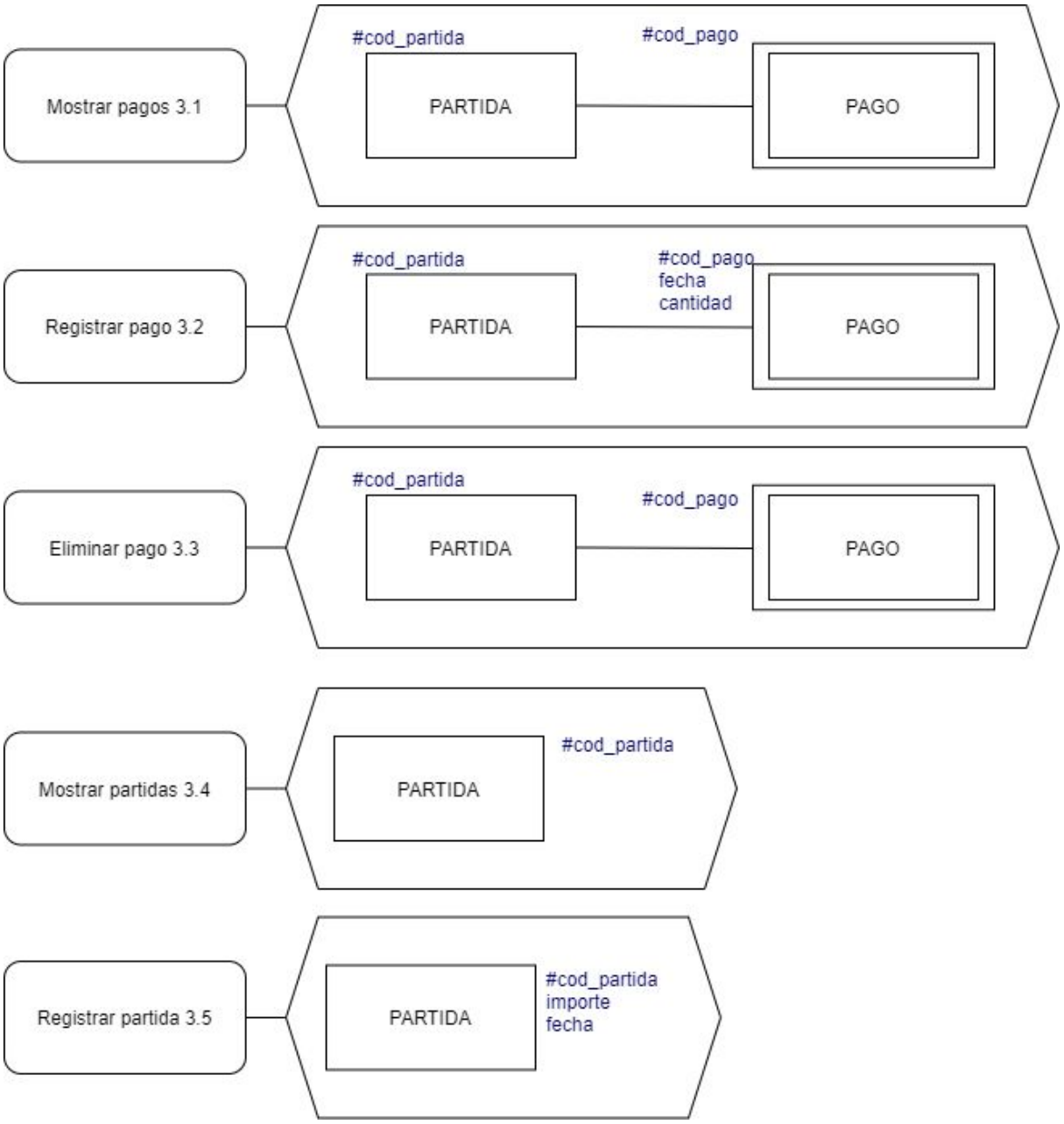
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Manuel Jose Cano Rojo (Proveedores)



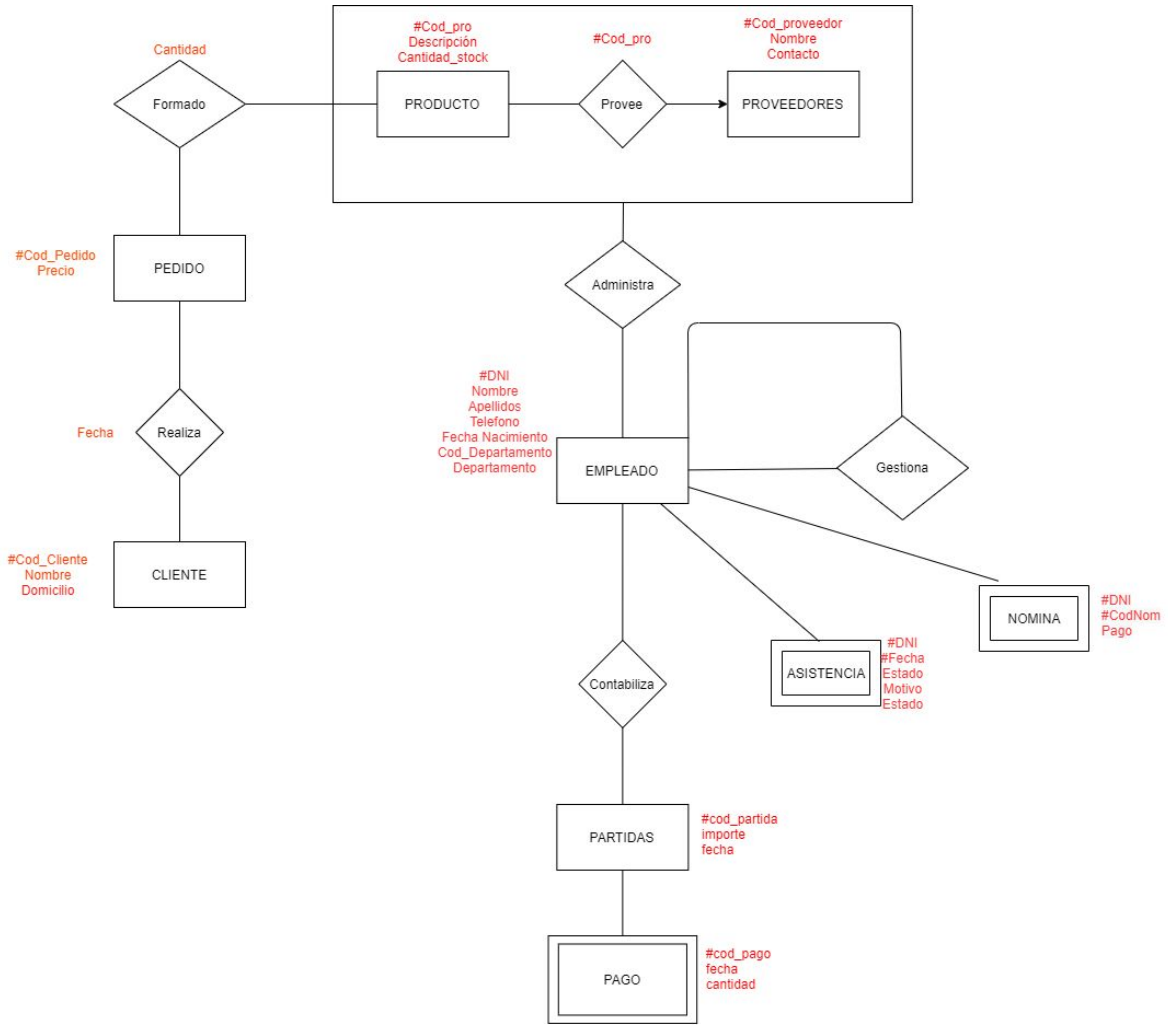
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

Santiago Carbó García (Finanzas)



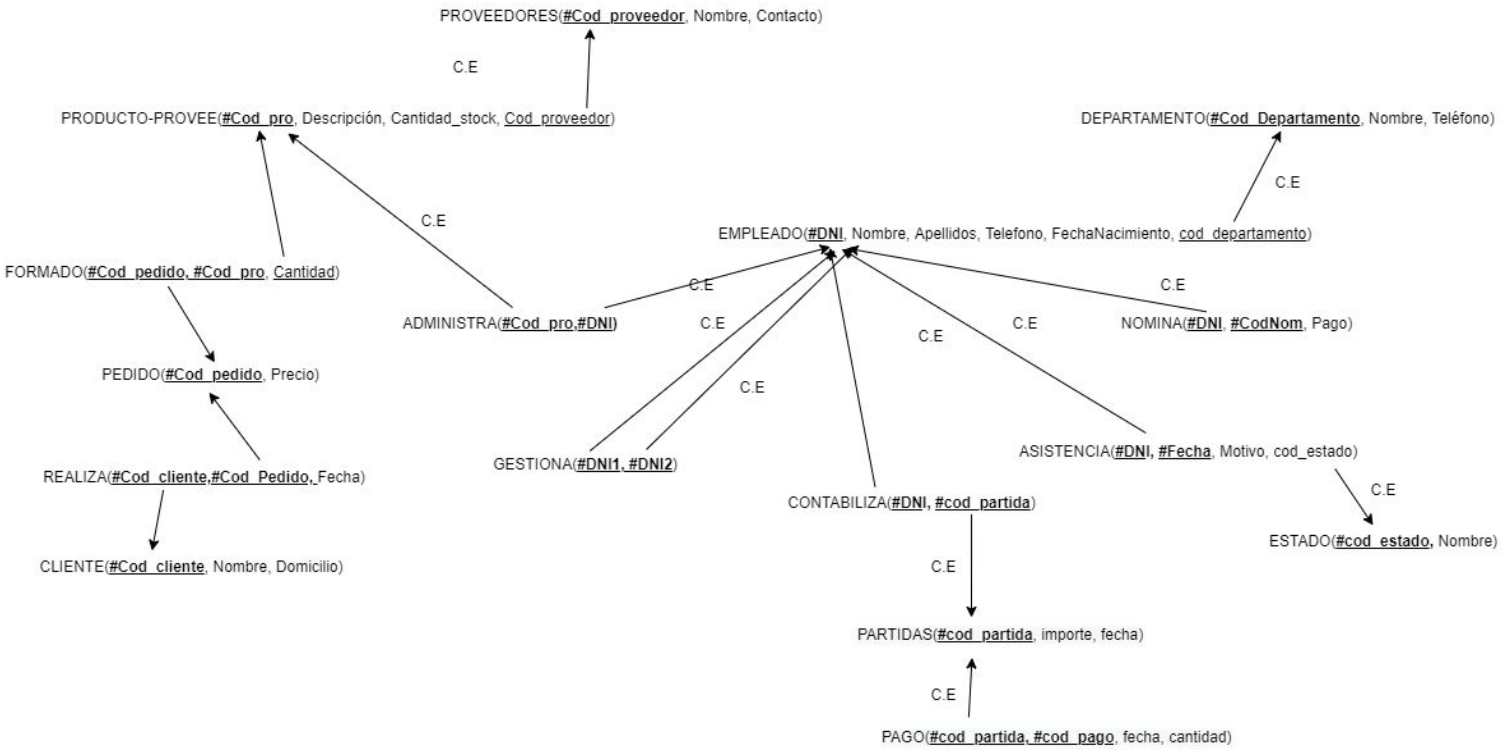
Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Diagrama Entidad Relación**



Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Paso a Tablas**



Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- ## DEPENDENCIAS FUNCIONALES ENCONTRADAS
    - **Producto:** todos los atributos depende funcionalmente #cod_pro.
    - **Proveedores:** todos los atributos depende funcionalmente de #DNI.
    - **Administra:** No tiene atributos no primos.
    - **Empleado:** Normalizamos la tabla empleado añadiendo cod_departamento y la entidad departamento.
    - **Formado:** los atributos no primos dependen de la clave primaria.
    - **Pedido:** los atributos no primos dependen de la clave primaria.
    - **Realiza:** los atributos no primos dependen de la clave primaria.
    - **Cliente:** los atributos no primos dependen de la clave primaria.
    - **Gestiona:** los atributos no primos dependen de la clave primaria.
    - **Contabiliza:** los atributos no primos dependen de la clave primaria.
    - **Partidas:** los atributos no primos dependen de la clave primaria.
    - **Pago:** los atributos no primos dependen de la clave primaria.
    - **Pertenece:** los atributos no primos dependen de la clave primaria.
    - **Departamento:** los atributos no primos dependen de la clave primaria.
    - **Nómina:** los atributos no primos dependen de la clave primaria.
    - **Asistencia:** Normalizamos añadiendo cod_estado a Asistencia y la entidad Estado.
    - **Tiene:** los atributos no primos dependen de la clave primaria.
    - **Estado:** los atributos no primos dependen de la clave primaria.

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **CREACIÓN DE TABLAS SQL**

```
create table PEDIDO(
cod_pedido varchar(20) primary key,
precio float(10) not null check (precio>=0)
);

create table CLIENTE(
cod_cliente varchar(20) primary key,
nombre char(20) NOT NULL,
domicilio varchar(40) NOT NULL
);

create table REALIZA(
cod_pedido references PEDIDO(cod_pedido),
cod_cliente references CLIENTE(cod_cliente),
fecha date,
PRIMARY KEY (cod_pedido,cod_cliente)
);

create table PRODUCTO(
cod_pro varchar(20) primary key,
descripción char(40),
cantidad_stock number(10) not null check (cantidad_stock>=0),
precio_pro float (10),
cod_proveedores varchar(20)
);

create table FORMADO(
cod_pedido references PEDIDO(cod_pedido),
cod_pro references PRODUCTO(cod_pro),
cantidad number(10) not null check (cantidad>=0),
PRIMARY KEY (cod_pedido, cod_pro)
);

create table DEPARTAMENTOS(
cod_departamento varchar(20) primary key,
nombre char(20) NOT NULL,
tlf number(9) NOT NULL
);
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
create table PROVEEDORES(
cod_proveedor varchar(20) primary key,
nombre char(20) NOT NULL,
contacto varchar(100) NOT NULL
);

create table PARTIDA(
cod_partida varchar(20) primary key,
importe float NOT NULL check (importe>=0),
fecha date NOT NULL
);

create table PAGO(
cod_partida varchar(20),
cod_pago varchar(20),
fecha date NOT NULL,
cantidad float NOT NULL check (cantidad>=0),
PRIMARY KEY (cod_partida, cod_pago),
FOREIGN KEY (cod_partida) REFERENCES partida(cod_partida)
);

create table EMPLEADO(
DNI varchar(20) primary key,
nombre char(20) NOT NULL,
apellidos char(20) NOT NULL,
tlf number(9) NOT NULL,
fecha_nacimiento date,
cod_departamento varchar(20) references DEPARTAMENTO(cod_departamento)
);

create table CONTABILIZA(
DNI references EMPLEADO(DNI),
cod_partida references PARTIDA(cod_partida),
PRIMARY KEY (DNI, cod_partida)
);


create table ADMINISTRA(
cod_pro references PRODUCTO(cod_pro),
DNI references EMPLEADO(DNI),
PRIMARY KEY (cod_pro,DNI)
);
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
create table GESTIONA(
DNI1 references EMPLEADO(DNI),
DNI2 references EMPLEADO(DNI)
);

create table NOMINA(
cod_nom varchar(20) primary key,
DNI references EMPLEADO(DNI),
pago float(9) NOT NULL
);

create table ASISTENCIA(
fecha date,
DNI varchar(20),
motivo varchar(300),
cod_estado varchar(20);
PRIMARY KEY (fecha,DNI),
FOREIGN KEY (DNI) references EMPLEADO(DNI)
);

create table ESTADO(
cod_estado varchar(20) primary key,
nombre varchar(20) not null
);
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **INSERCIÓN DE TUPLAS SQL**

```
insert into PEDIDO values ('cp2077',50);
insert into PEDIDO values ('cp2070',60);
insert into PEDIDO values ('cp2071',50);
insert into PEDIDO values ('cp2072',80);
insert into PEDIDO values ('cp2073',140);
insert into PEDIDO values ('cp2074',60);
insert into PEDIDO values ('cp2075',50);
insert into PEDIDO values ('cp2076',80);
insert into PEDIDO values ('cp2078',220);
insert into PEDIDO values ('cp2079',140);


insert into CLIENTE values ('77390380','Angel','calle 120');
insert into CLIENTE values ('77390381','Maxi','calle 121');
insert into CLIENTE values ('77390382','Maximiliano','calle 122');
insert into CLIENTE values ('77390383','Maximilian','calle 123');
insert into CLIENTE values ('77390384','Megamaxi','calle 124');
insert into CLIENTE values ('77390385','Maxitroll','calle 125');
insert into CLIENTE values ('77390386','Maximus','calle 126');
insert into CLIENTE values ('77390387','Minimaxi','calle 127');
insert into CLIENTE values ('77390388','Ultramaxi','calle 128');
insert into CLIENTE values ('77390389','Digimaxi','calle 129');


insert into REALIZA values ('cp2070','77390381',SYSDATE);
insert into REALIZA values ('cp2071','77390382',SYSDATE);
insert into REALIZA values ('cp2072','77390383',SYSDATE);
insert into REALIZA values ('cp2073','77390384',SYSDATE);
insert into REALIZA values ('cp2082','77390389',SYSDATE);


insert into PARTIDA values('pa200',3000,SYSDATE);
insert into PARTIDA values('pa210',3700,SYSDATE);
insert into PARTIDA values('pa220',2100,SYSDATE);
insert into PARTIDA values('pa230',10000,SYSDATE);
insert into PARTIDA values('pa240',2100,SYSDATE);
insert into PARTIDA values('pa300',3500,SYSDATE);
insert into PARTIDA values('pa310',7000,SYSDATE);
insert into PARTIDA values('pa320',800,SYSDATE);
insert into PARTIDA values('pa500',3200,SYSDATE);
insert into PARTIDA values('pa650',5400,SYSDATE);
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
insert into PAGO values('pa200','pg1',SYSDATE, 3000);
insert into PAGO values('pa210','pg1',SYSDATE, 3700);
insert into PAGO values('pa220','pg1',SYSDATE, 2100);
insert into PAGO values('pa230','pg1', SYSDATE, 10000);
insert into PAGO values('pa240','pg1',SYSDATE, 2100);
insert into PAGO values('pa300','pg1', SYSDATE, 3500);
insert into PAGO values('pa310','pg1', SYSDATE, 7000);
insert into PAGO values('pa320','pg1', SYSDATE, 800);
insert into PAGO values('pa500','pg1', SYSDATE, 3200);
insert into PAGO values('pa650','pg1', SYSDATE, 5400);
insert into PAGO values('pa650','pg2', SYSDATE, 5400);


insert into EMPLEADO values('22222222A', 'Jeremias', 'Amadeo que te
veo', 987654321, SYSDATE,2);
insert into EMPLEADO values('33333333A', 'Manolo', 'Parrilla Navarro',
321456987, SYSDATE,3);
insert into EMPLEADO values('44444444A', 'Paco', 'Roman Perez',
950328598, SYSDATE,4);
insert into EMPLEADO values('55555555A', 'Carlos', 'Segura Gonzalez',
971458214, SYSDATE,1);
insert into EMPLEADO values('66666666A', 'Santiago', 'Villegas Gomez',
958999888, SYSDATE,2);
insert into EMPLEADO values('77777777A', 'Jose', 'Perez Salmeron',
921456777, SYSDATE,3);
insert into EMPLEADO values('88888888A', 'Jorge', 'Gonzalez Ruiz',
952516666, SYSDATE,4);
insert into EMPLEADO values('99999999A', 'Daniel', 'Hernandez Gomez',
954774244, SYSDATE,1);
insert into EMPLEADO values('10101010A', 'Manuel', 'Hernandez Garcia',
950457144, SYSDATE,2);


insert into CONTABILIZA values('77777777A','pa670');
insert into CONTABILIZA values('22222222A','pa210');
insert into CONTABILIZA values('33333333A','pa220');
insert into CONTABILIZA values('44444444A','pa230');
insert into CONTABILIZA values('55555555A','pa240');
insert into CONTABILIZA values('22222222A','pa200');
insert into CONTABILIZA values('33333333A','pa300');
insert into CONTABILIZA values('44444444A','pa310');
insert into CONTABILIZA values('55555555A','pa320');
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
insert into CONTABILIZA values('22222222A','pa500');
insert into CONTABILIZA values('33333333A','pa650');
insert into CONTABILIZA values('44444444A','pa660');
insert into CONTABILIZA values('55555555A','pa680');
insert into CONTABILIZA values('22222222A','pa690');



insert into PROVEEDORES values ('pr1','Angel','suso@gmail.com');
insert into PROVEEDORES values ('pr2','Miguel','suso2@gmail.com');
insert into PROVEEDORES values ('pr3','Ignacion','suso3@gmail.com');
insert into PROVEEDORES values ('pr4','Andreu','suso4@gmail.com');
insert into PROVEEDORES values ('pr5','Mikel','suso5@gmail.com');
insert into PROVEEDORES values ('pr6','Jose','suso6@gmail.com');
insert into PROVEEDORES values ('pr7','Carlos','suso7@gmail.com');
insert into PROVEEDORES values ('pr8','Jose Miguel','suso8@gmail.com');
insert into PROVEEDORES values ('pr9','Clara','suso9@gmail.com');
insert into PROVEEDORES values ('pr10','Maria Jose',
'suso10@gmail.com');


insert into PRODUCTO values ('p1','Mayonesa','58',10,'pr1');
insert into PRODUCTO values ('p2','Aceitunas','43',20,'pr2');
insert into PRODUCTO values ('p3','Cebolla','65',30,'pr3');
insert into PRODUCTO values ('p4','Panteras Rosas','654',40,'pr4');
insert into PRODUCTO values ('p5','Anchoas','64',50,'pr5');
insert into PRODUCTO values ('p6','Queso Curado','16',60,'pr6');
insert into PRODUCTO values ('p7','Colonia','43',70,'pr7');
insert into PRODUCTO values ('p8','Crema','68',80,'pr8');
insert into PRODUCTO values ('p9','Coca Cola','104',90,'pr9');
insert into PRODUCTO values ('p10','Tomate frito', '234',100,'pr10');


insert into FORMADO values ('cp2082','p8',5);
insert into FORMADO values ('cp2078','p2',4);
insert into FORMADO values ('cp2079','p7',2);
insert into FORMADO values ('cp2070','p2',3);
insert into FORMADO values ('cp2071','p5',1);
insert into FORMADO values ('cp2072','p2',4);
insert into FORMADO values ('cp2073','p7',2);
insert into FORMADO values ('cp2074','p2',3);
insert into FORMADO values ('cp2075','p5',1);
insert into FORMADO values ('cp2076','p2',4);
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
insert into FORMADO values ('cp2078','p7',2);
insert into FORMADO values ('cp2078','p9',10);
insert into FORMADO values ('cp2078','p5',15);
insert into FORMADO values ('cp2080','p1',5);
insert into FORMADO values ('cp2081','p1',10);
insert into FORMADO values ('cp2081','p2',5);


insert into DEPARTAMENTOS values('1', 'Logistica', 1);
insert into DEPARTAMENTOS values('2', 'Recursos Humanos', 2);
insert into DEPARTAMENTOS values('3', 'Financiero', 3);
insert into DEPARTAMENTOS values('4', 'Proveedores', 4);


insert into GESTIONA values('22222222A', '11111111A');
insert into GESTIONA values('22222222A', '33333333A');
insert into GESTIONA values('22222222A', '44444444A');
insert into GESTIONA values('22222222A', '55555555A');
insert into GESTIONA values('22222222A', '66666666A');
insert into GESTIONA values('22222222A', '99999999A');
insert into GESTIONA values('99999999A', '22222222A');
insert into GESTIONA values('99999999A', '77777777A');
insert into GESTIONA values('99999999A', '88888888A');
insert into GESTIONA values('99999999A', '10101010A');


insert into NOMINA values('1', '11111111A', 2000);
insert into NOMINA values('2', '22222222A', 1500);
insert into NOMINA values('3', '33333333A', 1100);
insert into NOMINA values('4', '44444444A', 900);
insert into NOMINA values('5', '55555555A', 1300);
insert into NOMINA values('6', '66666666A', 1450);
insert into NOMINA values('7', '77777777A', 1200);
insert into NOMINA values('8', '88888888A', 1000);
insert into NOMINA values('9', '99999999A', 950);
insert into NOMINA values('10', '10101010A', 1325);


insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'11111111A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'22222222A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'33333333A', '');
```

```
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'44444444A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'55555555A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'66666666A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'77777777A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'88888888A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'99999999A', '');
insert into ASISTENCIA values(to_date('31/12/2020','dd/mm/yyyy'),
'10101010A', '');



insert into ESTADO values('1', 'Presente');
insert into ESTADO values('2', 'Retraso');
insert into ESTADO values('3', 'Ausente');
insert into ESTADO values('4', 'Falta Justificada');
insert into ESTADO values('5', 'Falta Injustificada');


insert into ADMINISTRA values ('p2','22222222A');
insert into ADMINISTRA values ('p3','33333333A');
insert into ADMINISTRA values ('p4','44444444A');
insert into ADMINISTRA values ('p5','55555555A');
```

- **MOTIVACIÓN PARA LA ELECCIÓN DEL SOFTWARE PARA DESARROLLAR LA APLICACIÓN**

El principal motivo que nos ha impulsado a usar Oracle SQL y Java con Netbeans, es que, estamos más familiarizados con este software, ya que, hemos estado trabajado con estos en otras asignaturas y consideramos que la información y las funcionalidades disponible acerca del mismo son perfectas para lo requerido este proyecto.

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- # TRANSACCIONES LÓGICAS
  - **Transacción Agregar Empleado:**

    “Insert” de un nuevo empleado:

```
"INSERT INTO EMPLEADO values('" + dni + "', '" + nombre + "', '" +
apellidos + "', '" +  teléfono + "', '"+ fecha_nacimiento + "', '"+
departamento +"'").
```

  - **Transacción Modificar Empleado:**

    En el caso de querer modificar el nombre del empleado: “update” del nombre de empleado:

```
"UPDATE EMPLEADO SET nombre='"+nombre+"' WHERE DNI= '"+dni+"'"
```

    En el caso de querer modificar los apellidos del empleado: “update” de los apellidos del empleado:

```
"UPDATE EMPLEADO SET apellidos='"+apellidos+"' WHERE DNI= '"+dni+"'"
```

    En el caso de querer modificar el contacto del empleado: “update” del contacto del empleado del empleado:

```
"UPDATE EMPLEADO SET tlf="+contacto+" WHERE DNI= '"+dni+"'"
```

    En el caso de querer modificar la fecha de nacimiento del empleado: “upda

```
"UPDATE EMPLEADO SET tlf="+contacto+" WHERE DNI= '"+dni+"'"
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Transacción Eliminar Empleado:**

  Elimina al empleado asociado con las tablas “ASISTENCIA”, “NOMINA”, “CONTABILIZA”, “ADMINISTRA”, “GESTIONA” y “EMPLEADO”:

```
"DELETE FROM ASISTENCIA WHERE DNI='"+dni+"'"
"DELETE FROM NOMINA WHERE DNI='"+dni+"'"
"DELETE FROM CONTABILIZA WHERE DNI='"+dni+"'"
"DELETE FROM ADMINISTRA WHERE DNI='"+dni+"'"
"DELETE FROM GESTIONA WHERE DNI1='"+dni+"' OR DNI2='"+dni+"'"
"DELETE FROM EMPLEADO WHERE DNI='"+dni+"'"
```

- **Transacción Registrar Pago:**

  “Insert” de un nuevo pago:

```
"INSERT INTO PAGO values('" + cod_partida + "', '" + cod_pago + "', '" +
fecha + "', '" +  cantidad + "')"
```

- **Transacción Registrar Pedido:**

  “Insert” de un nuevo pedido:

```
"INSERT INTO PEDIDO values('" + cod_pedido_nuevo + "', " + precio + ")"
```

A continuación, se realiza un “insert” y “update” por cada producto que queramos insertar:

```
"INSERT INTO FORMADO values('" + cod_pedido_nuevo + "', '" +
vcod_pro.get(i) + "', " + vcantidad.get(i) + ")"

"UPDATE PRODUCTO SET cantidad_stock=cantidad_stock-"+vcantidad.get(i)+"
WHERE cod_pro ='"+ vcod_pro.get(i) +"'"
```

Finalmente, introducimos un insert en la tabla “REALIZA”:

```
"INSERT INTO REALIZA values('" + cod_pedido_nuevo + "', '" + cod_cliente
```

```
+ "', SYSDATE)"
```

- **Transacción Modificar Pedido:**

  En primer lugar, eliminamos la fila con el pedido y el producto en concreto en "Formado":

```
"DELETE FROM FORMADO WHERE cod_pedido= '"+cod_pedido+"' AND
cod_pro='"+cod_pro+"'"
```

  Insertamos una nueva fila con la cantidad nueva:

```
"INSERT INTO FORMADO values('" + cod_pedido + "', '" + cod_pro + "', " +
cantidad + ")"
```

  Actualizamos la cantidad en "Producto":

  Si la cantidad es positiva, entonces estamos quitando de stock para ponerla en el pedido:

```
"UPDATE PRODUCTO SET cantidad_stock=cantidad_stock-"+cantidad_nueva+"
WHERE cod_pro='"+cod_pro+"'"
```

  Si la cantidad es negativa, entonces estamos añadiendo a stock:

```
"UPDATE PRODUCTO SET cantidad_stock=cantidad_stock+"+cantidad_nueva+"
WHERE cod_pro='"+cod_pro+"'"
```

  Actualizamos el precio en "Pedido":

```
"UPDATE PEDIDO SET precio=precio+"+precio_nuevo_st+" WHERE
cod_pedido='"+cod_pedido+"'"
```

- **Transacción Cancelar Pedido:**

En primer lugar solicitamos el código de pedido que queremos eliminar.

```
"SELECT cantidad, cod_pro FROM FORMADO WHERE cod_pedido='"+cod_pedido+"'"
```

Actualizamos la cantidad de stock que tiene ese pedido, para ello consultamos todos los productos que tiene ese pedido gracias a la tabla "formado" y actualizamos el stock de la tabla producto.

```
"UPDATE PRODUCTO SET cantidad_stock=cantidad_stock+"+cantidad+" WHERE
cod_pro = '"+cod_pro+"'"
```

A continuación eliminamos de "formado" y de "realiza" el pedido en cuestión, con nuestro "cod_pedido" que hemos solicitado.

```
"DELETE FROM FORMADO WHERE cod_pedido='"+cod_pedido+"'"
"DELETE FROM REALIZA WHERE cod_pedido='"+cod_pedido+"'"
```

Después de esto ya podemos eliminar el pedido en si pues no se encuentra en ningún otro sitio este código y así se cumplen las restricciones de integridad.

```
"DELETE FROM PEDIDO WHERE cod_pedido='"+cod_pedido+"'"
```

- **Transacción Registrar Proveedor:**

Insertamos un nuevo proveedor en la tabla "Proveedores":

```
"INSERT INTO PROVEEDORES values('" + cod_proveedor + "', '" + nombre +
"', '" + contacto + "')"
```

- **Transacción Modificar Proveedor:**

En el caso de que actualicemos el nombre del proveedor (de la tabla "Proveedores"):

```
"UPDATE PROVEEDORES SET nombre='"+nombre+"' WHERE
cod_proveedor='"+cod_proveedor+"'"
```

En el caso de que actualicemos el contacto del proveedor (de la tabla "Proveedores"):

```
"UPDATE PROVEEDORES SET contacto='"+contacto+"' WHERE
cod_proveedor='"+cod_proveedor+"'"
```

- **Transacción Eliminar Proveedor:**

  Eliminamos la fila que coincida con el código del proveedor de las tablas "Provee" y "Proveedores:

```
"DELETE FROM PROVEE WHERE cod_proveedor='"+cod_proveedor+"'"
"DELETE FROM PROVEEDORES WHERE cod_proveedor='"+cod_proveedor+"'"
```

- **Transacción Modificar Pago:**

```
"UPDATE PAGO SET cantidad="+cantidad+" WHERE cod_pago='"+ cod_pago +"'
AND cod_partida='"+cod_partida+"'"
```

- **Transacción Registrar Partida:**

```
"INSERT INTO PARTIDA values('" + cod_partida_nuevo + "', '" + importe +
"', SYSDATE)"
"INSERT INTO CONTABILIZA values('" + empleado + "', '" +
cod_partida_nuevo +"')"
```

- **Transacción Eliminar Partida:**

```
"DELETE FROM PAGO WHERE cod_partida='"+cod_partida+"'"
"DELETE FROM CONTABILIZA WHERE cod_partida='"+cod_partida+"'"
"DELETE FROM PARTIDA WHERE cod_partida='"+cod_partida+"'"
```

- **Transacción Modificar Partida:**

  Nos mostrará un menú donde nos indicará que queremos modificar de una partida, los cuales son:

  - **Cambiar Importe**

```
"UPDATE PARTIDA SET importe="+importe+" WHERE cod_partida
='"+cod_partida+"'"
```

  - **Modificar Pago**

```
ModificarPago(cod_partida);
```

  - **Añadir un Pago Nuevo**

```
RegistrarPago(cod_partida);
```

- **Terminar**
  `terminar = true;`

• FUNCIONALIDADES

- **Funcionalidad Subsistema de Recursos Humanos**

```
public void ModificarEmpleado()
{
    System.out.println("\nIntroduzca el DNI del Empleado");
    Scanner in = new Scanner (System.in);
    String dni = in.nextLine();
    int opcion;
    boolean terminar = false, error = false;
    Savepoint punto = null;

    try{
       punto = conexion.setSavepoint();

       while(!terminar){
          System.out.println("\nQue valor desea modificar
          \n1.Nombre.\n2.Apellidos.\n3.Contacto.\n4.Fecha de
          Nacimiento.\n5.Terminar.");
          opcion = Integer.parseInt(in.nextLine());

          switch(opcion){
             case 1:
                System.out.println("\nIntroduzca el nuevo Nombre
                (Introduzca -1 para cancelar la operacion)");
                String nombre = in.nextLine();
                if (!nombre.equals("-1"))
                   estado.executeUpdate("UPDATE EMPLEADO SET
                   nombre='"+nombre+"' WHERE DNI= '"+dni+"'");
             break;

             case 2:
                System.out.println("\nIntroduzca los nuevos Apellido
                (Introduzca -1 para cancelar la operacion)");
                String apellidos = in.nextLine();
                if (!apellidos.equals("-1"))
                   estado.executeUpdate("UPDATE EMPLEADO SET
                   apellidos='"+apellidos+"' WHERE DNI= '"+dni+"'");
             break;
```

```
            case 3:
               System.out.println("\nIntroduzca el nuevo Contacto
               (Introduzca -1 para cancelar la operacion)");
               String contacto = in.nextLine();
               if (!contacto.equals("-1"))
                  estado.executeUpdate("UPDATE EMPLEADO SET
                  tlf="+contacto+" WHERE DNI= '"+dni+"'");
            break;

            case 4:
               System.out.println("\nIntroduzca la nueva Fecha de
               Nacimiento DD/MM/YYYY (Introduzca -1 para cancelar la
               operacion)");
               String fecha_nacimiento = in.nextLine();
               if (!fecha_nacimiento.equals("-1"))
                  estado.executeUpdate("UPDATE EMPLEADO SET
                  fecha_nacimiento='"+fecha_nacimiento+"' WHERE DNI=
                  '"+dni+"'");
            break;

            case 5:
               terminar = true;
            break;
         }
      }
   }
   catch(SQLException ex){
      error = true;
      System.out.println("\nError al Modificar Empleado.\n");
      System.out.println("Error: "+ex.getMessage());
   }

   try{
      if (!error)
         conexion.commit();
      else
         conexion.rollback(punto);
   }
   catch(SQLException ex){
      System.out.println("\nError al hacer el commit o el rollback.\n");
      System.out.println("Error: "+ex.getMessage());
   }
}
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Funcionalidad Subsistema de Finanzas**

```
public void RegistrarPartida()
{
    Scanner in;
    boolean error = false;
    in = new Scanner (System.in);
    System.out.println("\nIntroduzca el empleado que contabiliza la
    partida\n");
    String empleado = in.nextLine();
    System.out.println("\nIntroduzca el importe de la partida.\n");
    String importe = in.nextLine();
    String numero_cod_partida_st = "", cod_partida;
    int numero_cod_partida, mayor = 0;
    Savepoint punto = null;

    try{
       punto = conexion.setSavepoint();
       ResultSet rs = estado.executeQuery("SELECT cod_partida FROM
       PARTIDA");

        while(rs.next()){
           cod_partida = rs.getString("cod_partida");
           numero_cod_partida_st =  cod_partida.replace("pa", "");
           numero_cod_partida = Integer.parseInt(numero_cod_partida_st);
           if (numero_cod_partida > mayor)
              mayor = numero_cod_partida;
           numero_cod_partida_st="";
        }

        mayor += 10;
        String cod_partida_nuevo = "pa"+mayor;

        System.out.println("\nAntes insert");
        estado.executeUpdate("INSERT INTO PARTIDA values('" +
        cod_partida_nuevo + "', '" + importe + "', SYSDATE)");

        System.out.println("\nDespues insert Partida");
        estado.executeUpdate("INSERT INTO CONTABILIZA values('" +
        empleado + "', '" + cod_partida_nuevo +"')");
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
        System.out.println("\nDespues insert Contabiliza");

        System.out.println("\n¿Desea Agregar Pagos a la
        partida?.\n1.Si.\n2.No.");
        int opcion  = Integer.parseInt(in.nextLine());
        if (opcion == 1){
           boolean terminar = false;
           while (!terminar){
              RegistrarPago(cod_partida_nuevo);
              System.out.println("\n¿Desea Agregar Otro Pago a la
              partida?.\n1.Si.\n2.No.");
              int opcion2  = Integer.parseInt(in.nextLine());
              if (opcion2 == 2)
                 terminar = true;
           }
        }

    }
    catch(SQLException ex){
       error = true;
       System.out.println("\nError al Registrar el Pedido.\n");
       System.out.println("Error: "+ex.getMessage());
    }

    try{
       if(!error){
          conexion.commit();;
          System.out.println("Partida Registrada con exito.");

       }
       else
          conexion.rollback(punto);
    }

    catch(SQLException ex){
          System.out.println("\nError en commit o en rollback.\n");
          System.out.println("Error: "+ex.getMessage());
     }
}
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Funcionalidad Subsistema de Logística**

```
public void ModificarPedido()
{
   System.out.println("\nIntroduzca el codigo del pedido");
   Scanner in = new Scanner (System.in);
   String cod_pedido = in.nextLine();
   boolean terminar = false, error = false;
   Savepoint punto = null;

   try{
      punto = conexion.setSavepoint();
      while(!terminar){
         System.out.println("\nIntroduzca el codigo del producto a
         añadir o modificar (Introduzca -1 para cancelar la
         operacion)");
         ResultSet rs = estado.executeQuery("SELECT * FROM FORMADO where
         cod_pedido='"+cod_pedido+"'");

         while (rs.next()){
            String cod_pro = rs.getString("cod_pro");
            System.out.println("\nCodigo Producto: "+cod_pro);
            System.out.println("Cantidad del producto:
            "+rs.getInt("cantidad"));
         }

         String cod_pro = in.nextLine();
         float precio;

         if (!cod_pro.equals("-1")){
            String cantidad_inicial;
            String cantidad_nueva, precio_nuevo;
            rs = estado.executeQuery("SELECT * FROM FORMADO where
            cod_pro='"+cod_pro+"' AND cod_pedido='"+cod_pedido+"'");

            if(rs.next())
               cantidad_inicial = rs.getString("cantidad");
            else
               cantidad_inicial = "0";

            System.out.println("\nIntroduzca la nueva cantidad del
```

```
            producto (La cantidad actual es "+cantidad_inicial+")");
            String cantidad = in.nextLine();

            estado.executeUpdate("DELETE FROM FORMADO WHERE cod_pedido=
            '"+cod_pedido+"' AND cod_pro='"+cod_pro+"'");

            if (!cantidad.equals("0"))
               estado.executeUpdate("INSERT INTO FORMADO values('" +
               cod_pedido + "', '" + cod_pro + "', " + cantidad + ")");




            rs = estado.executeQuery("SELECT * FROM PRODUCTO where
            cod_pro='"+cod_pro+"'");
            rs.next();
            precio = rs.getFloat("precio_pro");

            cantidad_nueva = "" + (Float.parseFloat(cantidad) -
            Float.parseFloat(cantidad_inicial));

            precio_nuevo = "" + precio*Float.parseFloat(cantidad_nueva);
            String precio_nuevo_st = "" + precio_nuevo;

            if (Float.parseFloat(cantidad_nueva) > 0)
               estado.executeUpdate("UPDATE PRODUCTO SET
               cantidad_stock=cantidad_stock-"+cantidad_nueva+" WHERE
               cod_pro='"+cod_pro+"'");

            else{
               cantidad_nueva = "" +
               Math.abs(Float.parseFloat(cantidad_nueva));

               estado.executeUpdate("UPDATE PRODUCTO SET
               cantidad_stock=cantidad_stock+"+cantidad_nueva+" WHERE
               cod_pro='"+cod_pro+"'");
            }

            estado.executeUpdate("UPDATE PEDIDO SET
            precio=precio+"+precio_nuevo_st+" WHERE
            cod_pedido='"+cod_pedido+"'");
            rs.close();

         }
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

```
            else
               terminar = true;
         }
      }

      catch(SQLException ex){
         error = true;
         System.out.println("\nError al Modificar Empleado.\n");
         System.out.println("Error: "+ex.getMessage());
      }

      try{
         if(!error)
            conexion.commit();
         else
            conexion.rollback(punto);

      }

      catch(SQLException ex){
            System.out.println("\nError al hacer el commit o el
            rollback.\n");
            System.out.println("Error: "+ex.getMessage());
      }
}
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García

- **Funcionalidad Subsistema de Proveedores**

```
Public void EliminarProveedor()
{
    System.out.println("\nIntroduzca el codigo del proveedor\n");
    Scanner in = new Scanner (System.in);
    String cod_proveedor = in.nextLine();
    Savepoint punto = null;
    boolean error = false;

    try{
        punto = conexion.setSavepoint();
        estado.executeUpdate("UPDATE PRODUCTO SET cod_proveedor=NULL
        WHERE cod_proveedor='"+cod_proveedor+"'");
        estado.executeUpdate("DELETE FROM PROVEEDORES WHERE
        cod_proveedor='"+cod_proveedor+"'");
    }

    catch(SQLException ex){
       error = true;
       System.out.println("\nError al borrar un proveedor de la tabla
       PROVEEDORES.\n");
       System.out.println("Error: "+ex.getMessage());
    }

    try{
       if (!error){
          conexion.commit();
          System.out.println("\nProveedor Eliminado con exito.\n");
       }
       else
          conexion.rollback(punto);
    }

    catch(SQLException ex){
        System.out.println("\nError al hacer commit o rollback en
        EliminarProveedor.\n");
    }

}
```

Manuel Jose Cano Rojo
Carlos García Segura
Angel Amadeo Gonzalez Ruiz
Santiago Carbó García