



UNIVERSIDAD DE GRANADA

Escuela Técnica Superior de Ingeniería Informática y Telecomunicaciones

Grupo 8 - DeliveryGranada

DDSI Prácticas D3

Juan Luis Torres Ramos

Mario Illán Valero

Alberto Rodríguez Lechuga

Thomas Lang

15 - Enero - 2024

Índice

1. Descripción del sistema	2
2. Áreas funcionales	3
3. Requisitos Funcionales, Restricciones semánticas	5
4. Esquema Caja Negra	38
5. DFD Armazón (DFD0)	38
6. DFD1	39
7. Esquemas Externos	43
8. Diagrama E/R	47
9. Extracción de tablas a partir E/R y Fusión de tablas	48
10. Justificación de Web Framework y SGBD	49
11. Scripts de creación de base de datos e inserción de datos	51
12. Conexión a la base de datos	59
13. Cursos,Disparador y excepción	60

1. Descripción del sistema

- **Nombre:** DeliveryGranada
- **Idea Principal:** Empresa de entrega de productos locales gastronómicos

La empresa DeliveryGranada gestiona la compra y entrega de productos comerciales locales a la puerta de los clientes. Los restaurantes, bares y establecimientos de comida locales pueden promocionar sus productos, bebidas, menús, platos y otros productos gastronómicos de manera económica con nuestro sistema.

Los clientes podrán visitar estos lugares para comprar sus productos favoritos o hacer sus compras diarias. El usuario puede administrar fácilmente sus pedidos a través de nuestra plataforma, que ofrece seguimiento en tiempo real e información detallada sobre cómo prepararlos.

DeliveryGranada se encarga de la gestión de pedidos, la atención al cliente, la contabilidad y el manejo de nuestros empleados. Nuestro objetivo es brindar a nuestros clientes y socios comerciales locales una experiencia fluida, brindándoles nuevas oportunidades de mercado.

En DeliveryGranada, nos esforzamos por conectar a los consumidores con los negocios locales y reconocemos la importancia de la responsabilidad ambiental. Para reducir la huella de carbono, nos esforzamos por utilizar embalajes eco amigables y optimizar las rutas de entrega.

Asimismo, estamos desarrollando programas de fidelización que fomentan la lealtad hacia los comercios locales participantes y recompensan a nuestros clientes habituales. Estos programas incluyen acceso anticipado a nuevas ofertas gastronómicas, ofertas exclusivas y descuentos.

2. Áreas funcionales

Human Resources HRIS

Person in charge: Thomas Lang

Description: The goal of human resources is an effective way of managing employees. It collects all the important data of the employees (contract details, work times, reports) to facilitate the work of the personnel department.

Goals:

- Management of new employees and be able to fire an employee
- Payroll: Organizes the payment of the workers
- Time and Attendance: Collects the work times, punctuality and sick days. connected to the payroll
- Training (LMS): Tracks qualifications, certifications or skills of the employees. Helps to recommend trainings to employees or to put employees in better fitting positions
- Benefit system: uses data to give benefits to workers who are punctual, loyal and/or efficient

Contabilidad

Encargado: Alberto Rodriguez

Descripción: Subsistema encargado de llevar las cuentas de la empresa teniendo en cuenta todas las actividades que suponen un gasto y los beneficios que se producen.

Objetivos:

- Gestión del cobro de nuestros socios comerciales.
- Gestión de las nóminas de los trabajadores.
- Gestión del gasto en publicidad y marketing.
- Balance de beneficios-gastos que se producen en la empresa.

Logística

Encargado: Mario Illán Valero

Descripción: Subsistema encargado de gestionar los pedidos de los compradores como la recepción, procesamiento y seguimiento del producto hasta que es entregado en el domicilio.

Objetivos:

- Recepción de los pedidos: se registran los pedidos mediante la aplicación y se verifica la disponibilidad de los productos en el inventario del negocio.
- Procesamiento de los pedidos: los pedidos se asignan a repartidores disponibles.
- Entrega del pedido: los repartidores recogen el pedido en el negocio y la app planifica la ruta más rápida y eficiente hacia el domicilio del cliente.
- Comunicación con el cliente: el cliente recibe notificaciones sobre el estado de las entregas, de la hora estimada de llegada, problemas con el pedido.
- Devoluciones y reclamaciones: manejo de devoluciones y reclamaciones de los clientes de manera eficiente.
- Seguimiento de los pedidos: el cliente podrá ver en tiempo real el seguimiento de todos los procesos por los que pasa su pedido hasta que llega a casa.

Cliente

Encargado: Juan Luis Torres Ramos

Descripción: Subsistema encargado de supervisar y administrar tanto a nuestros clientes, los compradores como a nuestros socios comerciales.

Objetivos:

- Se podrá manejar una entidad de usuario para el sistema. Incluye información personal, dirección y número de teléfono. Con esta información facilitamos futuros pedidos para los clientes.
- Se podrá manejar una entidad de restaurante para el sistema. Incluye Nombre de establecimiento, ubicación, tipo de negocio, etc
- Se podrá manejar los menús y los que el restaurante propicie.

3. Requisitos Funcionales, Restricciones semánticas

Subsistema 1: Human Resources HRIS

Person in charge: Thomas Lang

Requisito funcional			
Número de referencia	RF1.1		
Descripción del requisito	Add an employee		
Entrada	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	User have to insert the applicants data and subjective rating	
	Requisito de datos	Número de referencia	RDE1.1
		Composición	<ul style="list-style-type: none"> • Name: String (30) • Surname: String (30) • Email: String (30) • Phone number: Integer (15) • Direccion: String (30)
Datos internos	Requisito de datos	Número de referencia	RDW 1.1
		Composición	<ul style="list-style-type: none"> • Name: String (30) • Surname: String (30) • Email: String (30) • Phone number: Integer (15) • Direccion: String (30)
Salida	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	Confirms that all data was inserted and transmitted correctly	
	Requisito de datos	Número de referencia	RDS1.1
		Composición	<ul style="list-style-type: none"> • Name: String (30) • Surname: String (30) • Email: String (30) • Phone number: Integer (15) • Direccion: String (30)
Restricciones semánticas		Número de referencia	RS1.1
		Descripción	Applicants who are suitable to more than one position will get more than one entry
		Requisito funcional relacionado	RF1.1
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> • RDE1.1

Requisito funcional			
Número de referencia	RF1.2		
Descripción del requisito	Automates the calculation of the salary		
Entrada	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	Input the ID of the employee and the period which will be calculated	
	Requisito de datos	Número de referencia	RDE1.2
		Composición	<ul style="list-style-type: none"> Employee ID: Integer Starting Date: Date (dd.mm.yyyy) End Date: Date (dd.mm.yyyy)
Datos internos	Requisito de datos	Número de referencia	RDW 1.2
		Composición	<ul style="list-style-type: none"> Employee ID: Integer Starting Date: Date (dd.mm.yyyy) End Date: Date (dd.mm.yyyy) Efficiency RF 1.4
Salida	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	No action needed	
	Requisito de datos	Número de referencia	RDS1.2
		Composición	<ul style="list-style-type: none"> Surname of the employee: String (30) RF1.1 Name of the employee: String (30) RF1.1 Calculated salary: Integer IBAN: Integer RF 1.1
Restricciones semánticas		Número de referencia	RS1.2
		Descripción	If the end date of the period is before the starting date returns an error
		Requisito funcional relacionado	RF1.2
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDE1.2

Requisito funcional			
Número de referencia	RF1.3		
Descripción del requisito	Recording the worktime of the employee		
Entrada	Agente externo	Employee	
	Acción iniciada por el agente externo	The employee inserts his ID, number of made deliveries and the start and end time of his shift (possible to automate through an application).	
	Requisito de datos	Número de referencia	RDE1.3
		Composición	<ul style="list-style-type: none"> Employee ID: Integer Starting Date: Date (dd.mm.yyyy) End Date: Date (dd.mm.yyyy)
Datos internos	Requisito de datos	Número de referencia	RDW1.3
		Composición	<ul style="list-style-type: none"> Employee ID: Integer Starting Date: Date (dd.mm.yyyy) End Date: Date (dd.mm.yyyy) Surname of the employee: String (30) RF1.1 Name of the employee: String (30) RF1.1
Salida	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	No action needed	
	Requisito de datos	Número de referencia	RDS1.3
		Composición	<ul style="list-style-type: none"> Surname of the employee: String (30) RF1.1 Name of the employee: String (30) RF1.1 Calculated minutes: Integer Calculated Deliveries per hour: Integer
Restricciones semánticas		Número de referencia	RS1.3
		Descripción	If the end time is earlier than the starting time returns an error
		Requisito funcional relacionado	RF1.3
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDE1.3

Requisito funcional			
Número de referencia	RF1.4		
Descripción del requisito	Calculates the efficiency of the delivery drivers		
Entrada	Agente externo	Delivery Drivers	
	Acción iniciada por el agente externo	The driver just needs to input his ID	
	Requisito de datos	Número de referencia	RDE1.4
		Composición	<ul style="list-style-type: none"> Employee ID: Integer (4)
Datos internos	Requisito de datos	Número de referencia	RDW 1.4
		Composición	<ul style="list-style-type: none"> Employee ID: Integer (4) Surname of the employee: String (30) RF1.1 Name of the employee: String (30) RF1.1 Calculated deliveries per hour: Integer (3) RF1.3 Average reviews: integer (5) Calculated over all score
Salida	Agente externo	Personnel department / Administrator/ Delivery Driver	
	Acción iniciada por el agente externo	No action needed.	
	Requisito de datos	Número de referencia	RDS1.4
		Composición	<ul style="list-style-type: none"> Surname of the employee: String (30) RF1.1 Name of the employee: String (30) RF1.1 Calculated deliveries per hour: Integer Average reviews: integer/5 Calculated over all score
Restricciones semánticas		Número de referencia	RS1.4
		Descripción	Employee ID is not existend returns error
		Requisito funcional relacionado	RF1.1
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDE1.4

Requisito funcional			
Número de referencia	RF1.5		
Descripción del requisito	Remove an employee from the database		
Entrada	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	Put in the employee id	
	Requisito de datos	Número de referencia	RDE1.1
		Composición	<ul style="list-style-type: none"> Name: String (30)
Datos internos	Requisito de datos	Número de referencia	RDW 1.5
		Composición	<ul style="list-style-type: none"> Name: String (30) Email: String (30) Phone number: Int (15) Direccion: String (30) IBAN: chain of char (34) (4)
Salida	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	Confirm that everything was removed	
	Requisito de datos	Número de referencia	RDS1.5
		Composición	<ul style="list-style-type: none"> Name: String (30) Email: String (30) Phone number: Int (15) Direccion: String (30) IBAN: chain of char (34) (4)
Restricciones semánticas		Número de referencia	RS1.5
		Descripción	Employee doesn't exist
		Requisito funcional relacionado	RF1.5
		Requisito/s de datos relacionados	RDE1.5

Requisito funcional			
Número de referencia	RF1.6		
Descripción del requisito	Modify an employee		
Entrada	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	User have to insert the employees data you want to change	
	Requisito de datos	Número de referencia	RDE1.6
		Composición	<ul style="list-style-type: none"> Name: String (30) Surname: String(30) Email: String (30) Phone number: Int (15) Direccion: String (30) IBAN: chain of char (34) (4)
Datos internos	Requisito de datos	Número de referencia	RDW 1.7
		Composición	<ul style="list-style-type: none"> Name: String (30) Surname: String(30) Email: String (30) Phone number: Int (15) Direccion: String (30) IBAN: chain of char (34) (4)
Salida	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	Confirms that all data was inserted and transmitted correctly	
	Requisito de datos	Número de referencia	RDS1.6
		Composición	<ul style="list-style-type: none"> Name: String (30) Surname: String(30) Email: String (30) Phone number: Int (15) Direccion: String (30) IBAN: chain of char (34) (4)
Restricciones semánticas		Número de referencia	RS1.6
		Descripción	Employee doesn't exist
		Requisito funcional relacionado	RF1.6
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDE1.6

Requisito funcional			
Número de referencia	RF1.7		
Descripción del requisito	Show employee list		
Entrada	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo	User have to do nothing except running the function	
	Requisito de datos	Número de referencia	RDE1.7
		Composición	
Datos internos	Requisito de datos	Número de referencia	RDW 1.7
		Composición	<ul style="list-style-type: none"> • Name: String (30) • Surname: String(30) • Email: String (30) • Phone number: Int (15) • Direccion: String (30) • IBAN: chain of char (34) (4)
Salida	Agente externo	Personnel department / Administrator	
	Acción iniciada por el agente externo		
	Requisito de datos	Número de referencia	RDS1.7
		Composición	<ul style="list-style-type: none"> • Name: String (30) • Email: String (30) • Phone number: Int (15) • Direccion: String (30) • IBAN: chain of char (34) (4)
Restricciones semánticas		Número de referencia	
		Descripción	
		Requisito funcional relacionado	
		Requisito/s de datos relacionados	

Subsistema 2: Contabilidad

Encargado: Alberto Rodríguez Lechuga

Requisito funcional			
Número de referencia	RF2.1 - Dar de alta un ingreso.		
Descripción del requisito	Se registra un ingreso en el sistema, incluyendo detalles como la fecha y el importe.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Insertar el ingreso en el sistema	
	Requisito de datos	Número de referencia	RDE 2.1
		Composición	<ul style="list-style-type: none"> • Importe: Número con 2 decimales • Fecha: (DD/MM/YYYY)
Datos internos	Requisito de datos	Número de referencia	RDW 2.1
		Composición	<ul style="list-style-type: none"> • Importe: Número con 2 decimales • Fecha: (DD/MM/YYYY) • Comentario: cadena de texto
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación de alta de ingreso.	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 2.1
		Descripción	Si ya existía un ingreso con el mismo ID, no se inserta y se devuelve un error.
		Requisito funcional relacionado	RF 2.1.
		Requisito/s de datos relacionados	RDW 2.1

Requisito funcional			
Número de referencia	RF2.2 - Dar de baja un ingreso.		
Descripción del requisito	Permite la eliminación de un registro de un ingreso en el sistema, bien porque sea incorrecto o porque ya no sea relevante.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Eliminar un registro de un ingreso.	
	Requisito de datos	Número de referencia	RDE 2.2
		Composición	<ul style="list-style-type: none"> ID de la transacción: cadena de enteros (15)
Datos internos	Requisito de datos	Número de referencia	RDW 2.2
		Composición	<ul style="list-style-type: none"> Importe: Número con 2 decimales Fecha: (DD/MM/YYYY) Comentario: cadena de texto
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación de baja de ingreso	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 2.2
		Descripción	Si no existe un ingreso con el mismo ID, se devuelve un error.
		Requisito funcional relacionado	RF 2.2
		Requisito/s de datos relacionados	RDW 2.2

Requisito funcional			
Número de referencia	RF2.3 - Modificar un ingreso.		
Descripción del requisito	Permite actualizar algunos campos de un registro de ingreso existente.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Modificar un registro de ingreso.	
	Requisito de datos	Número de referencia	RDE 2.3
		Composición	<ul style="list-style-type: none"> ID de la transacción: cadena de enteros (15)
Datos internos	Requisito de datos	Número de referencia	RDW 2.3
		Composición	<ul style="list-style-type: none"> Fecha: (DD/MM/YYYY) Importe: Número con 2 decimales
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación de la modificación.	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 2.3
		Descripción	El importe del ingreso no puede reducirse después de registrarse.
		Requisito funcional relacionado	RF 2.3
		Requisito/s de datos relacionados	RDE 2.3 RDW 2.3

Requisito funcional			
Número de referencia	RF2.4 - Listar una serie de ingresos.		
Descripción del requisito	Se listan una serie de ingresos almacenados en el sistema con unos requisitos específicos, lo que facilita la revisión y el análisis de los ingresos.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Listar unos ingresos con ciertos requisitos.	
	Requisito de datos	Número de referencia	RDE 2.4
		Composición	<ul style="list-style-type: none"> Fecha (DD/MM/YYYY)
Datos internos	Requisito de datos	Número de referencia	RDR 2.4
		Composición	<ul style="list-style-type: none"> Importe: Número con 2 decimales Fecha: (DD/MM/YYYY) Comentario: cadena de texto
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación del listado de ingresos.	
	Requisito de datos	Número de referencia	RDS 2.4
		Composición	<ul style="list-style-type: none"> Importe: Número con 2 decimales Fecha: (DD/MM/YYYY) Comentario: cadena de texto
Restricciones semánticas		Número de referencia	RS 2.4
		Descripción	La lista de ingresos debe estar ordenada de forma predeterminada por fecha en orden ascendente, lo que permite ver los ingresos más antiguos primero.
		Requisito funcional relacionado	RF 2.4
		Requisito/s de datos relacionados	RDR 2.4 RDS 2.4

Requisito funcional			
Número de referencia	RF2.5 - Dar de alta un gasto.		
Descripción del requisito	Se registra un gasto en el sistema, incluyendo detalles como la fecha y el importe.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Insertar un gasto en el sistema.	
	Requisito de datos	Número de referencia	RDE 2.5
		Composición	<ul style="list-style-type: none"> • Importe: Número con 2 decimales • Fecha: (DD/MM/YYYY)
Datos internos	Requisito de datos	Número de referencia	RDW 2.5
		Composición	<ul style="list-style-type: none"> • Importe: Número con 2 decimales • Fecha: (DD/MM/YYYY) • Comentario: cadena de texto
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación de alta de gasto.	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 2.5
		Descripción	Si ya existía un gasto con el mismo ID, no se inserta y se devuelve un error.
		Requisito funcional relacionado	RF 2.5
		Requisito/s de datos relacionados	RDW 2.5

Requisito funcional			
Número de referencia	RF2.6 - Dar de baja un gasto.		
Descripción del requisito	Permite la eliminación de un registro de un gasto en el sistema, bien porque sea incorrecto o porque ya no sea relevante.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Eliminar un registro de un gasto.	
	Requisito de datos	Número de referencia	RDE 2.6
		Composición	<ul style="list-style-type: none"> ID de la transacción: cadena de enteros (15)
Datos internos	Requisito de datos	Número de referencia	RDW 2.6
		Composición	<ul style="list-style-type: none"> Importe: Número con 2 decimales Fecha: (DD/MM/YYYY) Comentario: cadena de texto
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación de baja de gasto	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 2.6
		Descripción	Si no existe un gasto con el mismo ID, se devuelve un error.
		Requisito funcional relacionado	RF 2.6
		Requisito/s de datos relacionados	RDW 2.6

Requisito funcional			
Número de referencia	RF2.7 - Modificar un gasto.		
Descripción del requisito	Permite actualizar algunos campos de un registro de gasto existente.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Modificar un registro de gasto.	
	Requisito de datos	Número de referencia	RDE 2.7
		Composición	<ul style="list-style-type: none"> ID de la transacción: cadena de enteros (15)
Datos internos	Requisito de datos	Número de referencia	RDW 2.7
		Composición	<ul style="list-style-type: none"> Fecha: (DD/MM/YYYY) Importe: Número con 2 decimales
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación de la modificación.	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 2.7
		Descripción	La fecha del gasto no puede ser cambiada por una fecha anterior a la fecha original del registro.
		Requisito funcional relacionado	RF 2.7
		Requisito/s de datos relacionados	RDE 2.7 RDW 2.7

Requisito funcional			
Número de referencia	RF2.8 - Listar una serie de gastos.		
Descripción del requisito	Se listan una serie de gastos almacenados en el sistema con unos requisitos específicos, lo que facilita la revisión y el análisis de los gastos.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Listar unos gastos con unos requisitos específicos.	
	Requisito de datos	Número de referencia	RDE 2.8
		Composición	<ul style="list-style-type: none"> Fecha (DD/MM/YYYY)
Datos internos	Requisito de datos	Número de referencia	RDR 2.8
		Composición	<ul style="list-style-type: none"> Importe: Número con 2 decimales Fecha: (DD/MM/YYYY) Comentario: cadena de texto
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirmación del listado de gastos.	
	Requisito de datos	Número de referencia	RDS 2.8
		Composición	<ul style="list-style-type: none"> Importe: Número con 2 decimales Fecha: (DD/MM/YYYY)
Restricciones semánticas		Número de referencia	RS 2.8
		Descripción	La lista de gastos debe estar ordenada de forma predeterminada por fecha en orden ascendente, lo que permite ver los gastos más antiguos primero
		Requisito funcional relacionado	RF 2.8
		Requisito/s de datos relacionados	RDR 2.8 RDS 2.8

Subsistema 3: Logística

Encargado: Mario Illán Valero

Requisito funcional			
Número de referencia	RF3.1 - Registrar pedido		
Descripción del requisito	Solicitar inserción de un nuevo pedido en el sistema		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar inserción de los datos de un nuevo pedido	
	Requisito de datos	Número de referencia	RDE 3.1
		Composición	<ul style="list-style-type: none"> Nombre de producto: cadena de caracteres (30) Cantidad: número entero positivo Dirección de entrega: cadena de caracteres (50) Datos del cliente: RDE4.1
Datos internos	Requisito de datos	Número de referencia	RDW 3.1
		Composición	<ul style="list-style-type: none"> Nombre de producto: cadena de caracteres (30) Cantidad: número entero positivo Dirección de entrega: cadena de caracteres (50) Fecha: date Código Identificador del pedido: cadena de enteros (10) Precio del pedido: cadena de enteros (20)
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	Confirmar pedido asignándole un identificador	
	Requisito de datos	Número de referencia	RDS 3.1
		Composición	<ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10) Dirección: cadena de caracteres (50) Estado del pedido: string
Restricciones semánticas		Número de referencia	RS 3.1
		Descripción	Si uno de los productos solicitados no se encuentra disponible en el almacén del negocio, se le informará al cliente con un aviso
		Requisito funcional relacionado	RF 3.1
		Requisito/s de datos relacionados	RDW 3.1

Requisito funcional			
Número de referencia	RF3.2 - Cancelar pedido		
Descripción del requisito	Cancelación del pedido		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar borrado/cancelación de un pedido mediante el ID	
	Requisito de datos	Número de referencia	RDE 3.2
		Composición	<ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDW 3.2
		Composición	<ul style="list-style-type: none"> Estado del pedido: string Fecha. date
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	Confirma el resultado de la cancelación del pedido	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 3.2
		Descripción	Un pedido que se haya preparado y asignado a un repartidor no se puede cancelar
		Requisito funcional relacionado	RF 3.2 RF 3.4
		Requisito/s de datos relacionados	RDW 3.2

Requisito funcional			
Número de referencia	RF3.3 - Consultar datos del pedido		
Descripción del requisito	Consulta los productos solicitados en su pedido		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicita información sobre el pedido mediante el ID del pedido	
	Requisito de datos	Número de referencia	RDE 3.3
		Composición	<ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDR 3.3
		Composición	<ul style="list-style-type: none"> Nombre de producto: cadena de caracteres (30) Cantidad: número entero positivo Dirección de entrega: cadena de caracteres (50) Estado del pedido: string Fecha estimada de entrega: date
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	Confirma la llegada de la información del pedido	
	Requisito de datos	Número de referencia	RDS 3.3
		Composición	<ul style="list-style-type: none"> Nombre de producto: cadena de caracteres (30) Cantidad: número entero positivo Dirección de entrega: cadena de caracteres (50) Estado del pedido: string Fecha estimada de entrega: date
Restricciones semánticas		Número de referencia	RS 3.3
		Descripción	Aunque se encuentre en estado "Cancelado" se podrá consultar la información de ese pedido
		Requisito funcional relacionado	RF 3.2
		Requisito/s de datos relacionados	RDR 3.3

Requisito funcional			
Número de referencia	RF3.4 - Cambiar estado del pedido		
Descripción del requisito	Modifica el estado del pedido. Son 5 estados: En Preparación, Preparado, En Reparto, Entregado, Cancelado.		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Asigna un nuevo estado al pedido	
	Requisito de datos	Número de referencia	RDE 3.4
		Composición	<ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDW 3.4
		Composición	<ul style="list-style-type: none"> Estado del pedido: string Fecha: date
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma el cambio de estado	
	Requisito de datos	Número de referencia	
		Composición	
Restricciones semánticas		Número de referencia	RS 3.4
		Descripción	Un mismo pedido con su identificador no puede tener más de un estado asociado
		Requisito funcional relacionado	RF 3.4
		Requisito/s de datos relacionados	RDW 3.4

Requisito funcional			
Número de referencia	RF3.5 - Notificar estado del pedido		
Descripción del requisito	Se notifica el cambio de estado del pedido al usuario		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Informa al usuario de su pedido cuando éste cambie de estado	
	Requisito de datos	Número de referencia	RDE 3.5
		Composición	<ul style="list-style-type: none"> • Código Identificador del pedido: cadena de enteros (10) • Estado del pedido: string • Fecha estimada de entrega: date
Datos internos	Requisito de datos	Número de referencia	RDR 3.5
		Composición	<ul style="list-style-type: none"> • Estado del pedido: string • Fecha estimada de entrega: date
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	Confirma el resultado	
	Requisito de datos	Número de referencia	RDS 3.5
		Composición	<ul style="list-style-type: none"> • Código Identificador del pedido: cadena de enteros (10) • Estado del pedido: string • Fecha estimada de entrega: date
Restricciones semánticas		Número de referencia	RS 3.5
		Descripción	Sólo se puede notificar el estado de un pedido a un usuario cuando hay un código identificador de pedido asignado a este.
		Requisito funcional relacionado	RF3.1
		Requisito/s de datos relacionados	RDW 3.1

Requisito funcional			
Número de referencia	RF3.6 - Asignar pedido		
Descripción del requisito	Se asigna el pedido a un repartidor para su reparto		
Entrada	Agente externo	Administrador	
	Acción iniciada por el agente externo	Asigna el pedido a un repartidor	
	Requisito de datos	Número de referencia	RDE 3.6
		Composición	<ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDW 3.6 RDR 3.6
		Composición	RDW 3.6 <ul style="list-style-type: none"> Código identificador de repartidor: cadena de enteros (15) RDR 3.6 <ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10) Estado del pedido: string Dirección del negocio: cadena de caracteres (40) Datos del cliente: RDE4.1
Salida	Agente externo	Repartidor	
	Acción iniciada por el agente externo	Confirma la llegada de la información del pedido	
	Requisito de datos	Número de referencia	RDS 3.6
		Composición	<ul style="list-style-type: none"> Código Identificador del pedido: cadena de enteros (10) Dirección del negocio: cadena de caracteres (40) Datos del cliente: RDE 4.1
Restricciones semánticas		Número de referencia	RS 3.6
		Descripción	Sólo se puede asignar un pedido por cada repartidor
		Requisito funcional relacionado	RF 3.6
		Requisito/s de datos relacionados	RDR 3.6

Requisito funcional			
Número de referencia	RF3.9 - Dar de alta un producto		
Descripción del requisito	Permite introducir un producto		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar la inserción de los datos de un nuevo producto	
	Requisito de datos	Número de referencia	RDE 3.9
		Composición	<ul style="list-style-type: none"> Nombre del producto: cadena de caracteres (100) Precio del producto: número con decimales (8)
Datos internos	Requisito de datos	Número de referencia	RDW 3.9
		Composición	<ul style="list-style-type: none"> Código identificador del menú: cadena de enteros (10) Código Identificador del producto: cadena de enteros (10) Nombre del producto: cadena de caracteres (100) Precio del producto: número con decimales (8)
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma la inserción del producto	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 3.9
		Descripción	No puede haber 2 productos con un mismo identificador
		Requisito funcional relacionado	RF 3.9
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDW 3.9

Requisito funcional			
Número de referencia	RF3.10 - Dar de baja un producto		
Descripción del requisito	Permite eliminar un producto existente		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar la eliminación de un producto	
	Requisito de datos	Número de referencia	RDE 3.10
		Composición	<ul style="list-style-type: none"> Código Identificador del producto: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDW 3.10
		Composición	Los mismos que RDW 3.9
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma que se ha borrado el producto correctamente	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 3.10
		Descripción	Sólo se puede eliminar un producto existente
		Requisito funcional relacionado	RF 3.9
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDW 3.9

Requisito funcional				
Número de referencia	RF3.11 - Modificar un producto			
Descripción del requisito	Permite modificar los datos de un producto existente			
Entrada	Agente externo	Restaurante		
	Acción iniciada por el agente externo	Solicitar la modificación de los datos de un producto existente		
	Requisito de datos	Número de referencia	RDE 3.11	
		Composición	<ul style="list-style-type: none">Código Identificador del producto: cadena de enteros (10)	
Datos internos	Requisito de datos	Número de referencia	RDW 3.11	
		Composición	Los mismos que RDW 3.9	
Salida	Agente externo	Administrador		
	Acción iniciada por el agente externo	Confirma la modificación de un producto		
	Requisito de datos	Número de referencia	ninguno	
		Composición		
Restricciones semánticas		Número de referencia	RS 3.11	
		Descripción	Sólo se puede modificar un producto existente y con datos concordantes con los RDW correspondientes	
		Requisito funcional relacionado	RF 3.11	
		Requisito/s de datos relacionados	<ul style="list-style-type: none">RDS 3.11, RDW 3.9	

Requisito funcional			
Número de referencia	RF3.12 - Consultar un producto		
Descripción del requisito	Permite consultar un producto		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar la información de un producto con cierto identificador	
	Requisito de datos	Número de referencia	RDE 3.12
		Composición	<ul style="list-style-type: none"> Código Identificador del producto: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDR 3.12
		Composición	Los mismos que RDW 3.9
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma y muestra el producto	
	Requisito de datos	Número de referencia	RDS 3.12
		Composición	Los mismos que RDW 3.9
Restricciones semánticas		Número de referencia	RS 3.12
		Descripción	Sólo se puede consultar los datos de un producto existente
		Requisito funcional relacionado	RF 3.12
		Requisito/s de datos relacionados	RDW 3.9

Subsistema 4: Cliente

Encargado: Juan Luis Torres Ramos

Requisito funcional			
Número de referencia	RF 4.1		
Descripción del requisito	Dar de alta al usuario		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar inserción de los datos de un nuevo usuario	
	Requisito de datos	Número de referencia	RDE 4.1
		Composición	<ul style="list-style-type: none"> • nombre: Cadena caracteres (255) • apellidos: cadena caracteres (255) • DNI: Cadena caracteres (9) • teléfono: Cadena caracteres (9) • dirección: Cadena caracteres (255)
Datos internos	Requisito de datos	Número de referencia	RDW 4.1
		Composición	<ul style="list-style-type: none"> • nombre: Cadena caracteres (255) • apellidos: cadena caracteres (255) • DNI: Cadena caracteres (9) • teléfono: Cadena caracteres (9) • dirección: Cadena caracteres (255)
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	Confirmar resultado.	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.1
		descripción	No se permite que dos usuarios tengan el mismo DNI en el sistema. Cada DNI debe ser único en la base de datos. Si ya ha había un usuario con el mismos DNI, no se inserta el nuevo contacto y se devuelve error
		requisito funcional relacionado	RF 4.1
		requisito de datos relacionados	RDW 4.1

Requisito funcional			
Número de referencia	RF 4.2		
Descripción del requisito	Dar de baja usuario		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar borrado de los datos de un usuario	
	Requisito de datos	Número de referencia	RDE 4.2
		Composición	<ul style="list-style-type: none"> DNI: Cadena caracteres (9)
Datos internos	Requisito de datos	Número de referencia	RDW 4.2
		Composición	<ul style="list-style-type: none"> Los mismos datos que RDW 4.1
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	Confirmar resultado	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.2
		Descripción	Si no hay DNI en la base de datos, se devuelve el error de que no existe dicho usuario.
		requisito funcional relacionado	RF 4.2
		requisito de datos relacionados	RDW 4.2

Requisito funcional			
Número de referencia	RF 4.3		
Descripción del requisito	Modificar datos del usuario		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar modificación de los datos de un usuario	
	Requisito de datos	Número de referencia	RDE 4.3
		Composición	<ul style="list-style-type: none"> los mismos que RDE 4.1
Datos internos	Requisito de datos	Número de referencia	RDW 4.3
		Composición	<ul style="list-style-type: none"> nombre: Cadena caracteres (20) teléfono: Cadena caracteres (15) dirección: Cadena caracteres (30)
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	confirmar resultados	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.3
		Descripción	Los datos introducidos tienen que ser coherentes en la base de datos, tanto el teléfono y la dirección serán comprobados para que existan, sino el sistema devuelve error
		requisito funcional relacionado	RF 4.3
		requisito de datos relacionado	RDW 4.3

Requisito funcional			
Número de referencia	RF 4.4		
Descripción del requisito	Consultar datos de usuarios		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar datos de un usuario para visualizarlo	
	Requisito de datos	Número de referencia	RDE 4.4
		Composición	los mismos que RDE 4.2
Datos internos	Requisito de datos	Número de referencia	RDR 4.4
		Composición	los mismos que RDW 4.1
Salida	Agente externo	Usuario	
	Acción iniciada por el agente externo	confirmar resultados	
	Requisito de datos	Número de referencia	RDS 4.3
		Composición	los mismos que RDR 4.4
Restricciones semánticas		Número de referencia	RS 4.3
		Descripción	Si no hay DNI en la base de datos, se devuelve el error de que no existe dicho usuario para consultar datos.
		requisito funcional relacionado	RF 4.4
		requisito de datos relacionado	RDE 4.4 + RDR 4.4

Requisito funcional			
Número de referencia	RF 4.5		
Descripción del requisito	Dar de alta restaurante		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar inserción de los datos de un nuevo restaurante	
	Requisito de datos	Número de referencia	RDE 4.5
		Composición	<ul style="list-style-type: none"> • nombre: Cadena caracteres (255) • propietario: Cadena caracteres (20) • dirección: Cadena caracteres (255) • teléfono: Cadena caracteres (9) • Número de registro comercial: Cadena caracteres (15) • empleados: Lista (dinámica)
Datos internos	Requisito de datos	Número de referencia	RDW 4.5
		Composición	<ul style="list-style-type: none"> • Los mismos datos que RDE 4.5
Salida	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Confirmar resultado.	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.5
		Descripción	No se permite que dos usuarios tengan el mismo número de registro comercial en el sistema. Si ya hay un restaurante con el mismo registro comercial, no se inserta el nuevo restaurante y se devuelve error
		requisito funcional relacionado	RF 4.5
		requisito de datos relacionado	RDW 4.5

Requisito funcional			
Número de referencia	RF 4.6		
Descripción del requisito	Dar de baja restaurante		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar borrado de los datos de un usuario	
	Requisito de datos	Número de referencia	RDE 4.6
		Composición	<ul style="list-style-type: none"> Número de registro comercial: Cadena caracteres (15)
Datos internos	Requisito de datos	Número de referencia	RDW 4.2
		Composición	<ul style="list-style-type: none"> Los mismos datos que RDW 4.5
Salida	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Confirmar resultado	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.6
		Descripción	Si no hay registro comercial en la base de datos, se devuelve el error que no existe dicho restaurante
		requisito funcional relacionado	RF 4.6
		requisito de datos relacionado	RDW 4.2

Requisito funcional			
Número de referencia	RF 4.7		
Descripción del requisito	Modificar datos del restaurante		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar modificación de los datos del restaurante	
	Requisito de datos	Número de referencia	RDE 4.7
		Composición	<ul style="list-style-type: none"> los mismos que RDE 4.5
Datos internos	Requisito de datos	Número de referencia	RDW 4.7
		Composición	<ul style="list-style-type: none"> nombre: Cadena caracteres (20) propietario: Cadena caracteres (20) dirección: Cadena caracteres (20) teléfono: Cadena caracteres (15) empleados: Lista (dinámica)
Salida	Agente externo	Restaurante	
	Acción iniciada por el agente externo	confirmar resultados	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.7
		Descripción	Los datos introducidos tienen que ser coherentes en la base de datos, sino el sistema devuelve error
		requisito funcional relacionado	RF 4.7
		requisito de datos relacionado	RDW 4.7

Requisito funcional			
Número de referencia	RF 4.8		
Descripción del requisito	Consultar datos de restaurante		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar datos de un restaurante para visualizarlo	
	Requisito de datos	Número de referencia	RDE 4.8
		Composición	los mismos que RDE 4.5
Datos internos	Requisito de datos	Número de referencia	RDR 4.8
		Composición	los mismos que RDW 4.5
Salida	Agente externo	Restaurante	
	Acción iniciada por el agente externo	confirmar resultados	
	Requisito de datos	Número de referencia	RDS 4.8
		Composición	los mismos que RDW 4.5
Restricciones semánticas		Número de referencia	RS 4.8
		Descripción	Si no esta el registro comercial en la base de datos, se devuelve el error de que no existe dicho restaurante para consultar datos
		requisito funcional relacionado	RF 4.8
		requisito de datos relacionado	RDE 4.8 + RDR 4.8

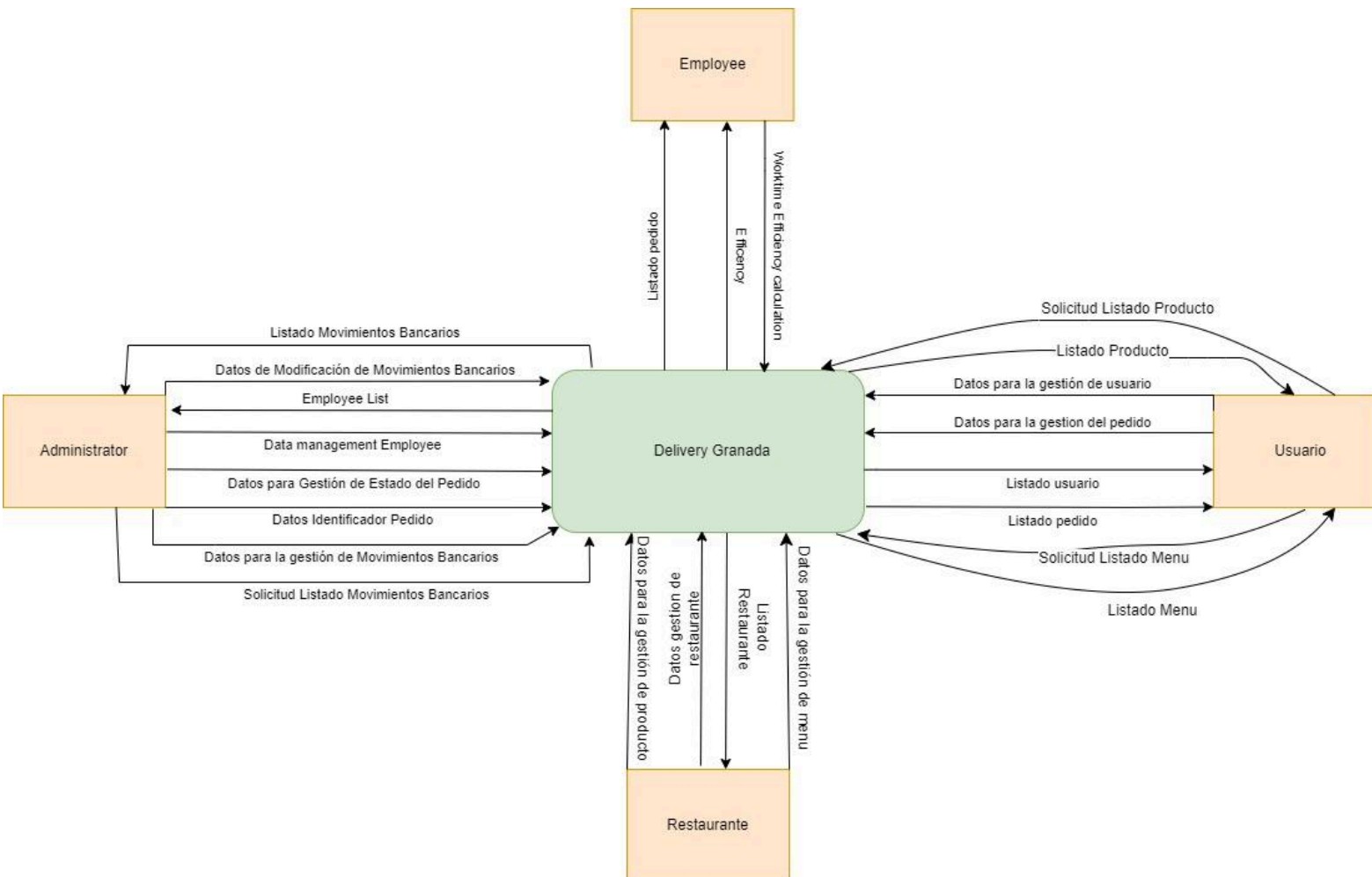
Requisito funcional			
Número de referencia	RF4.9 - Dar de alta un menú		
Descripción del requisito	Permite introducir un menú de un restaurante		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar la inserción de los datos de un nuevo menú	
	Requisito de datos	Número de referencia	RDE 4.9
		Composición	<ul style="list-style-type: none"> NRC del restaurante: cadena de caracteres (15) Nombre del menú: cadena de caracteres (15)
Datos internos	Requisito de datos	Número de referencia	RDW 4.9
		Composición	<ul style="list-style-type: none"> Código Identificador del menú: cadena de enteros (10) NRC del restaurante: cadena de caracteres (15) Nombre del menú: cadena de caracteres (15) Nombre del restaurante: cadena de caracteres (15)
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma que se ha insertado el menú correctamente	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.9
		Descripción	Un menú no puede estar en 2 restaurantes diferentes a la vez y no puede haber 2 menús distintos con el mismo identificador.
		Requisito funcional relacionado	RF 4.9
		Requisito/s de datos relacionados	RDW 4.9

Requisito funcional			
Número de referencia	RF4.10 - Dar de baja un menú		
Descripción del requisito	Permite eliminar un menú de un restaurante		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar el borrado de los datos de un menú existente	
	Requisito de datos	Número de referencia	RDE 4.10
		Composición	<ul style="list-style-type: none"> Código Identificador del menú: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDW 4.10
		Composición	Los mismos que RDW 4.9
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma que se ha borrado el menú correctamente	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.10
		Descripción	Sólo se puede eliminar un menú que exista.
		Requisito funcional relacionado	RF 4.10
		Requisito/s de datos relacionados	<ul style="list-style-type: none"> RDW 4.10

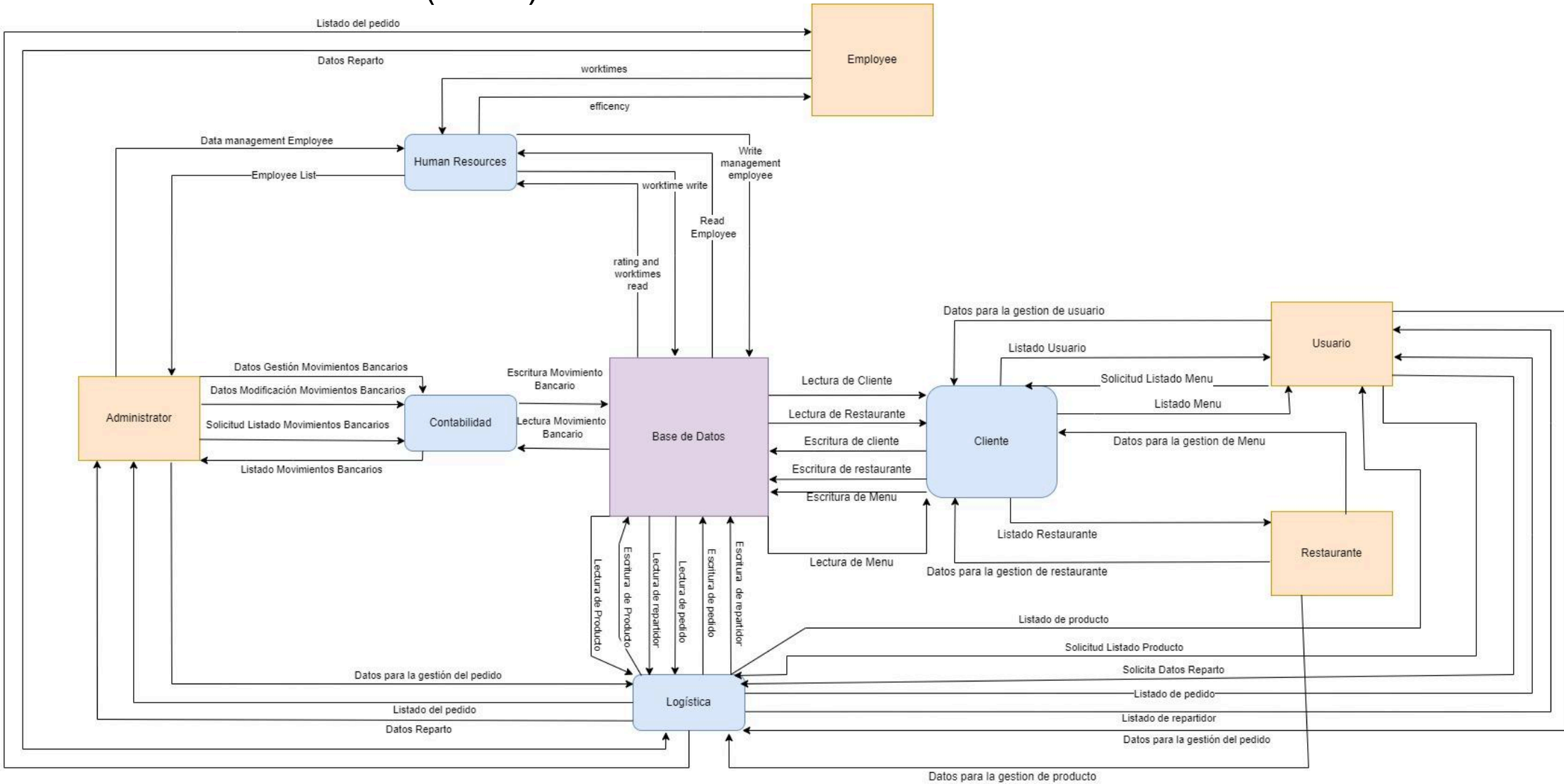
Requisito funcional			
Número de referencia	RF4.11 - Modificar un menú		
Descripción del requisito	Permite modificar un menú existente		
Entrada	Agente externo	Restaurante	
	Acción iniciada por el agente externo	Solicitar la modificación de los datos de un menú existente	
	Requisito de datos	Número de referencia	RDE 4.11
		Composición	<ul style="list-style-type: none"> Código Identificador del menú: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDW 4.11
		Composición	Los mismos que RDW 4.9
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma la modificación del menú	
	Requisito de datos	Número de referencia	ninguno
		Composición	
Restricciones semánticas		Número de referencia	RS 4.11
		Descripción	Sólo se puede modificar un menú existente y con datos concordantes con los RDW correspondientes
		Requisito funcional relacionado	RF 4.11
		Requisito/s de datos relacionados	RDW 4.9, RDS 4.11

Requisito funcional			
Número de referencia	RF4.12 - Consultar un menú		
Descripción del requisito	Permite consultar un menú		
Entrada	Agente externo	Usuario	
	Acción iniciada por el agente externo	Solicitar los datos del menú con cierto identificador	
	Requisito de datos	Número de referencia	RDE 4.12
		Composición	<ul style="list-style-type: none"> Código Identificador del menú: cadena de enteros (10)
Datos internos	Requisito de datos	Número de referencia	RDR 4.12
		Composición	Los mismos que RDW 4.9
Salida	Agente externo	Administrador	
	Acción iniciada por el agente externo	Confirma y muestra el menú	
	Requisito de datos	Número de referencia	RDS 4.12
		Composición	<ul style="list-style-type: none"> Código Identificador del menú: cadena de enteros (10) NRC del restaurante: cadena de caracteres (15) Nombre del menú: cadena de caracteres (15) Nombre del restaurante: cadena de caracteres (15)
Restricciones semánticas		Número de referencia	RS 4.12
		Descripción	Sólo se puede consultar los datos de un menú existente
		Requisito funcional relacionado	RF 4.12
		Requisito/s de datos relacionados	RDW 4.9

4. Esquema Caja Negra



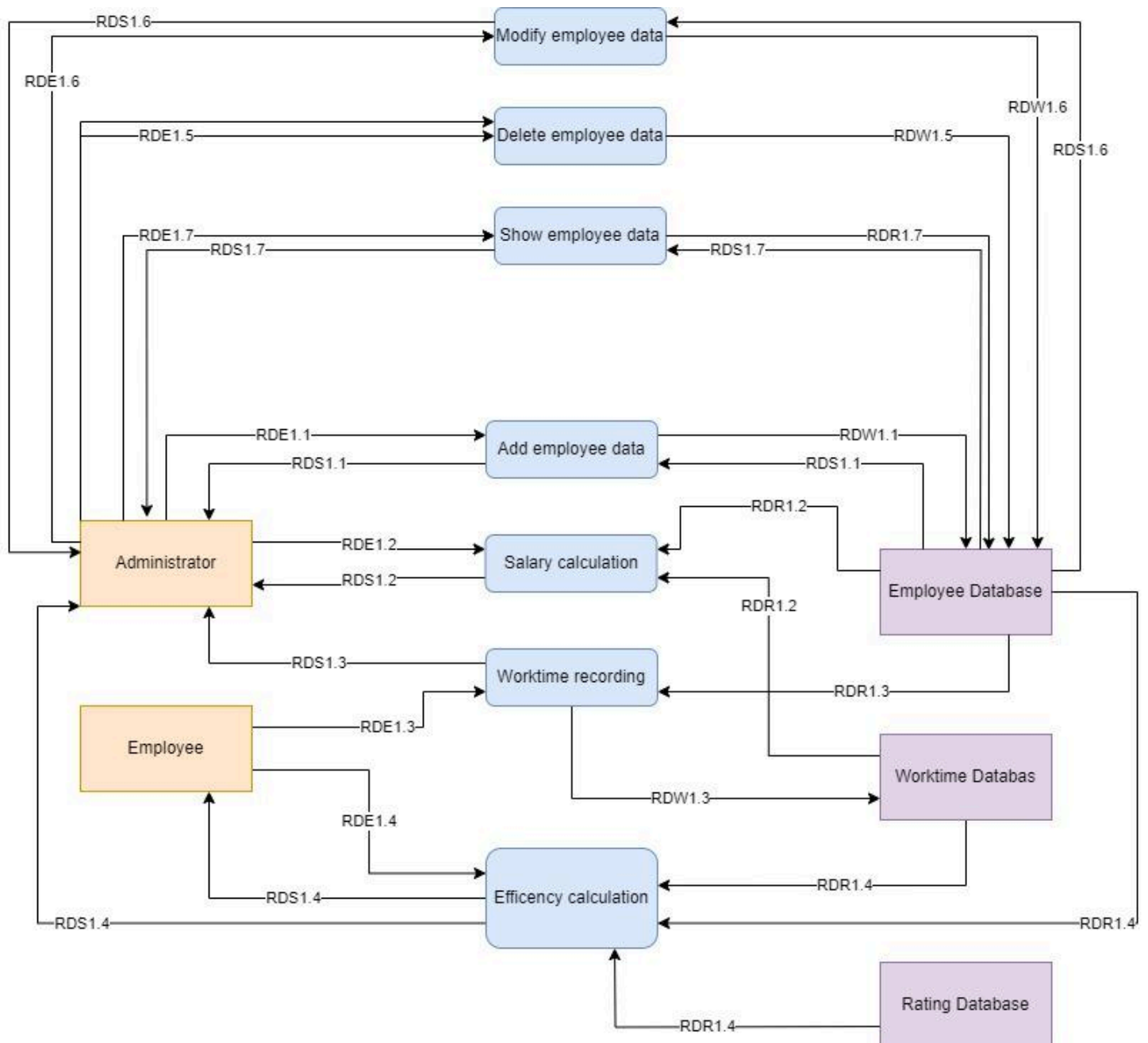
5. DFD Armazón (DFD0)



6. DFD1

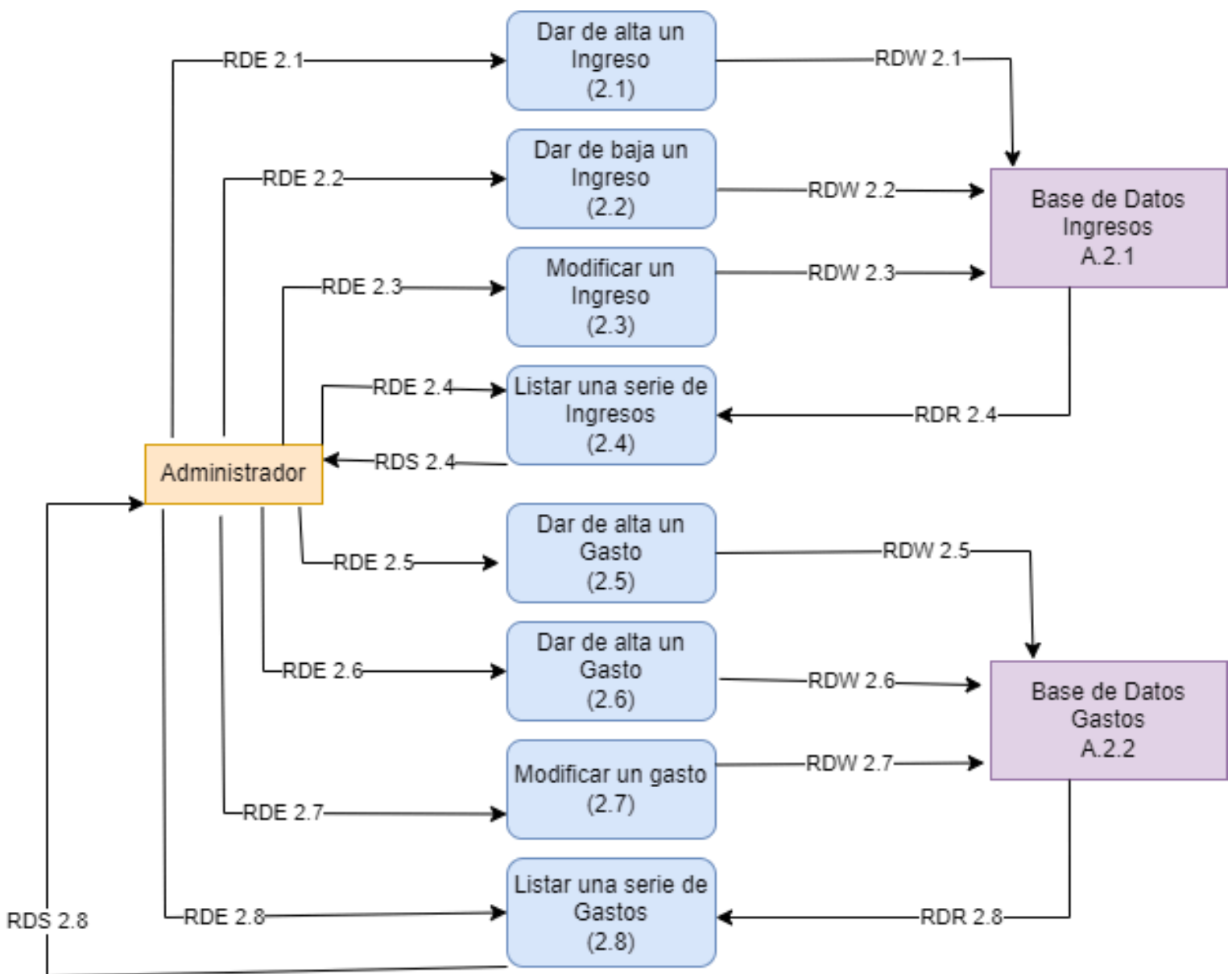
6.1 Human Resources

Person in charge: Thomas Lang

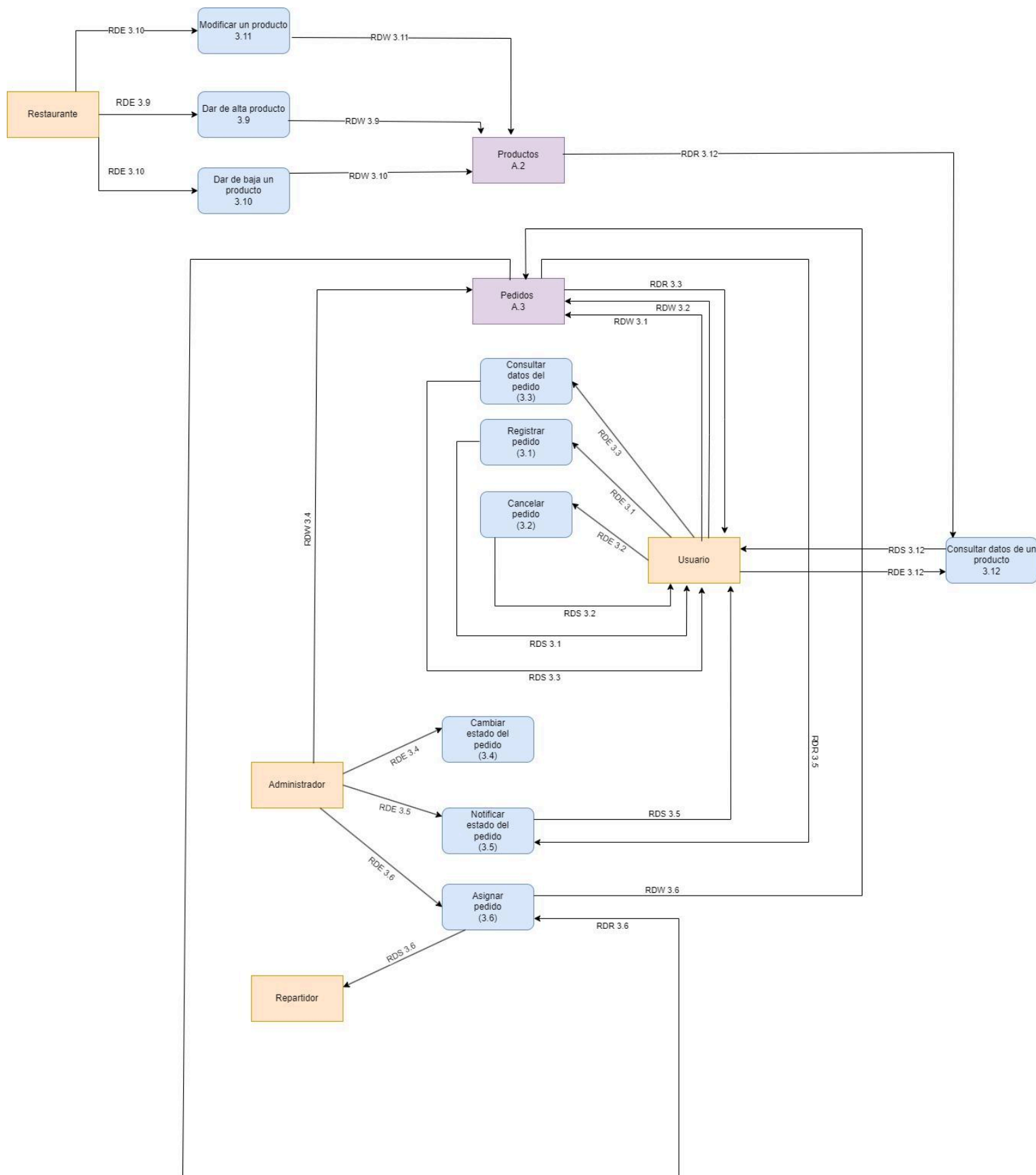


6.2 Contabilidad

Encargado: Alberto Rodríguez Lechuga

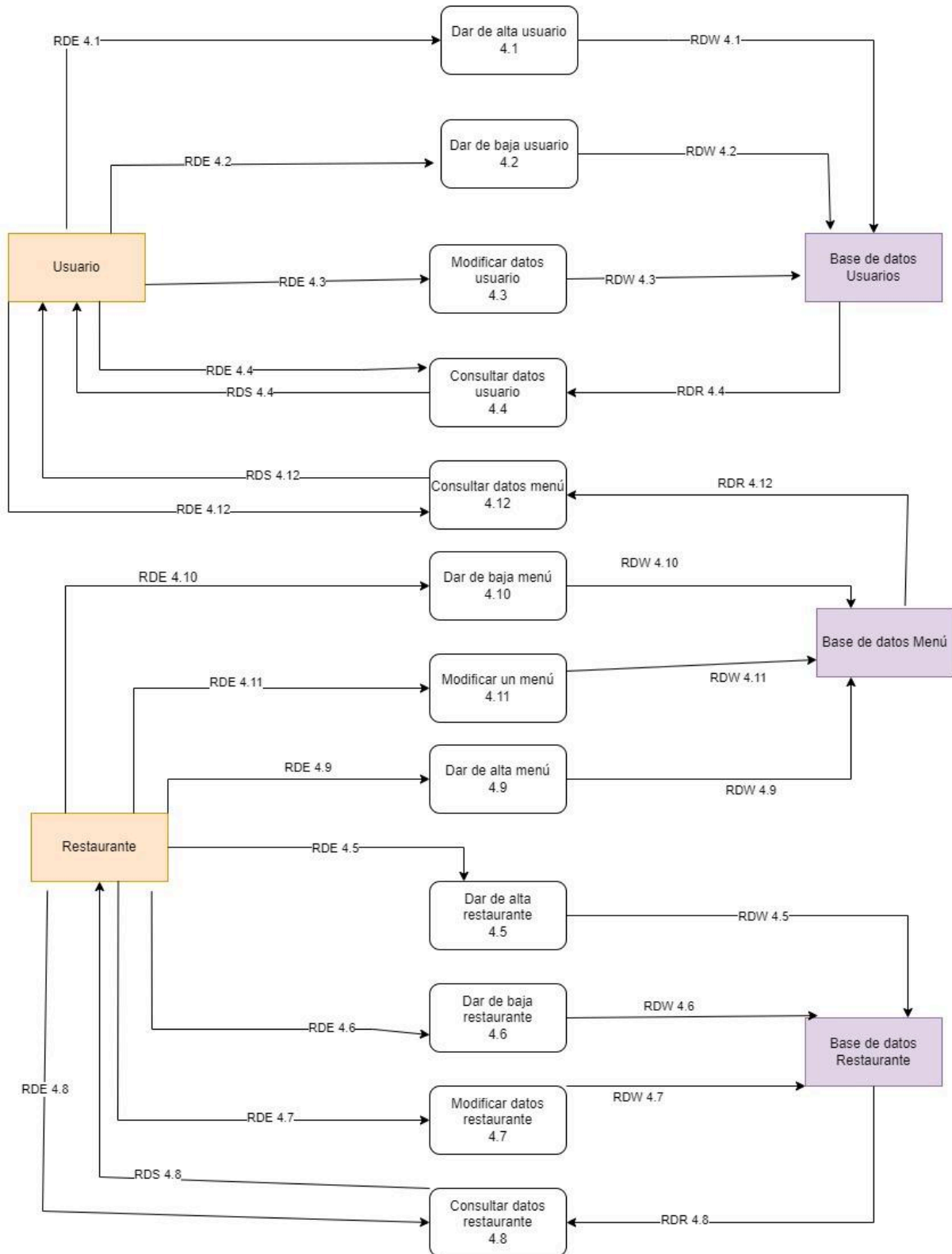


Encargado: Mario Illán Valero



6.4 Cliente

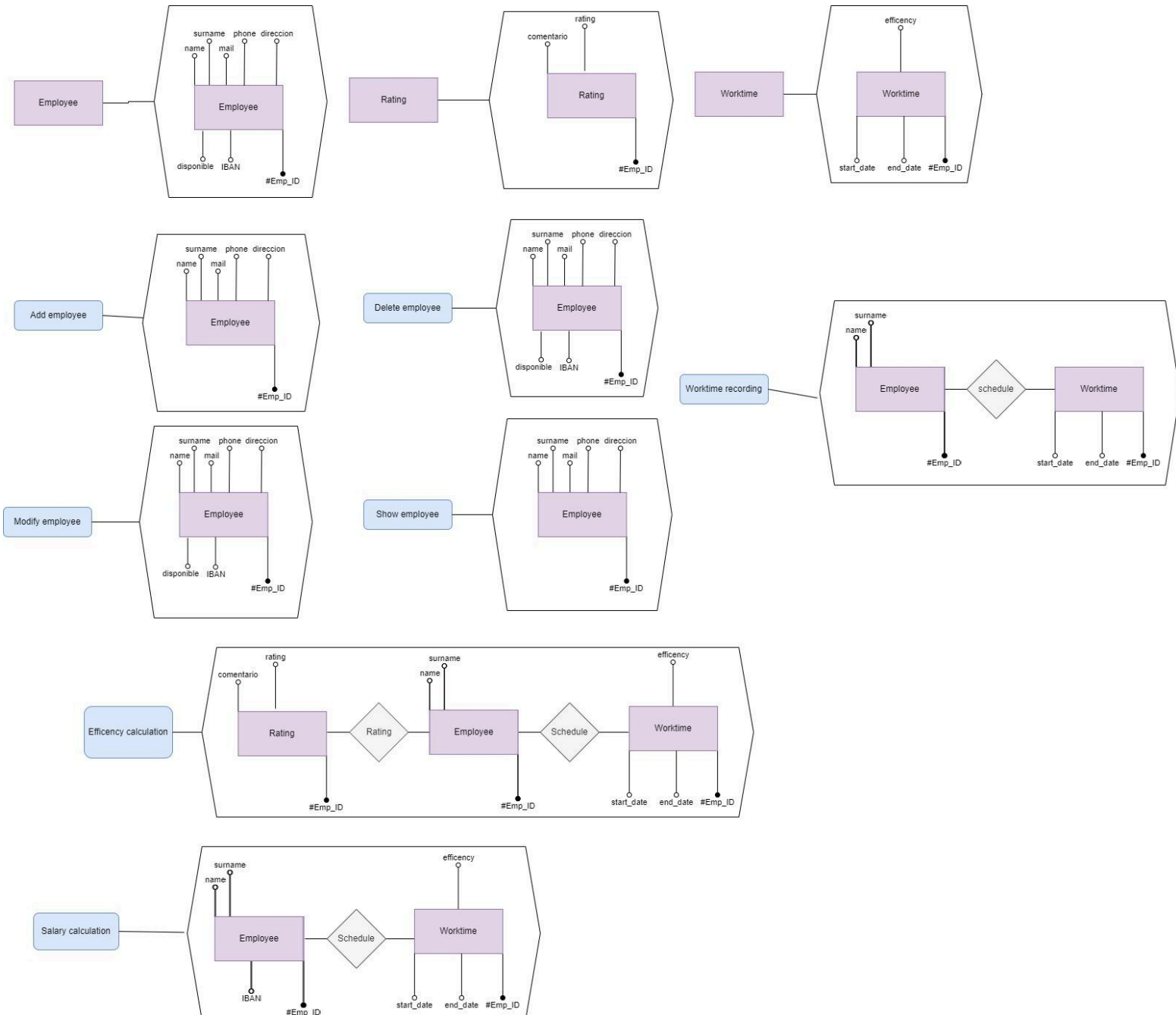
Encargado: Juan Luis Torres Ramos



7. Esquemas Externos

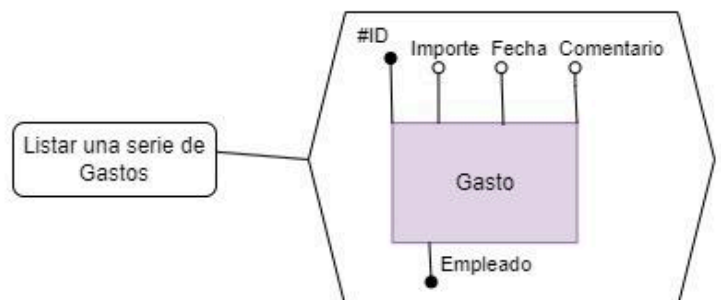
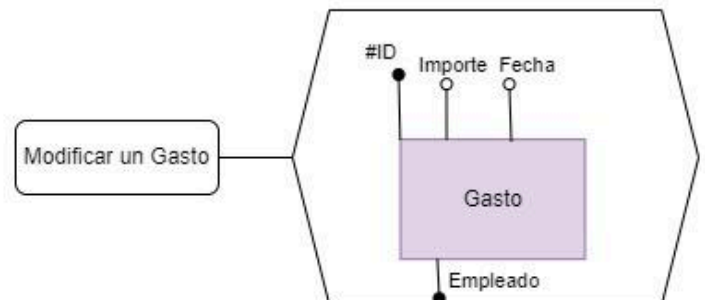
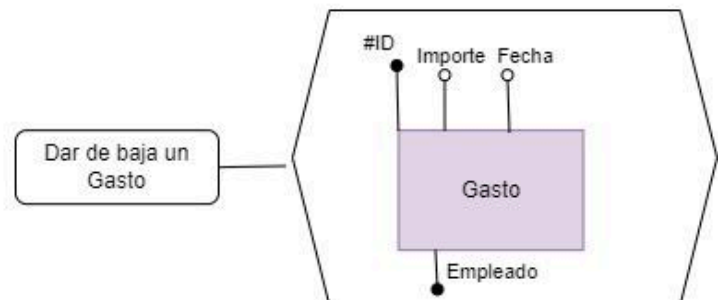
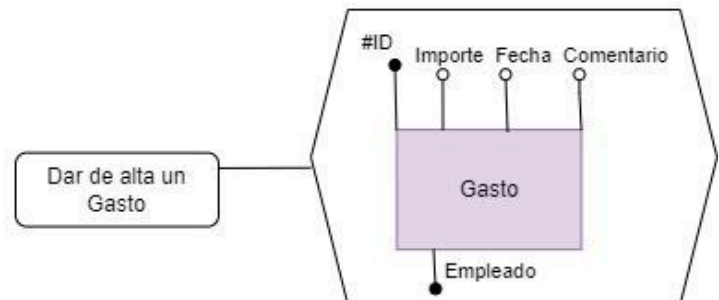
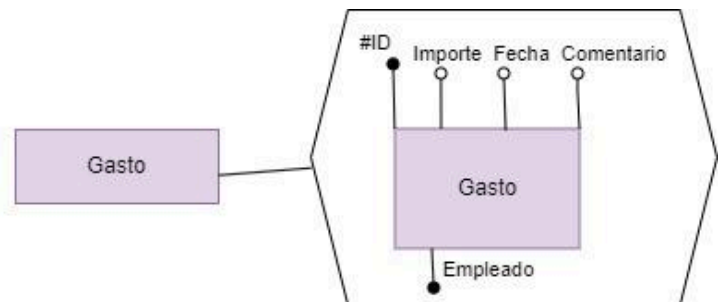
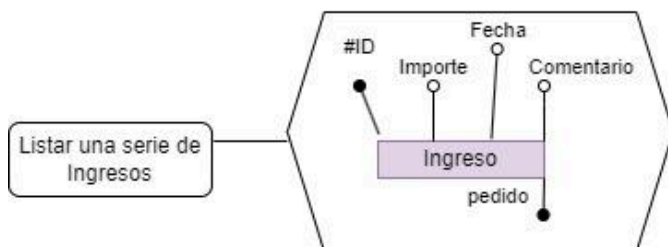
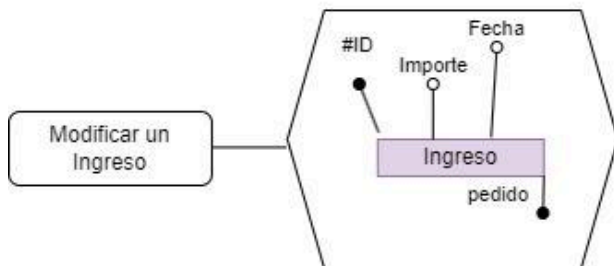
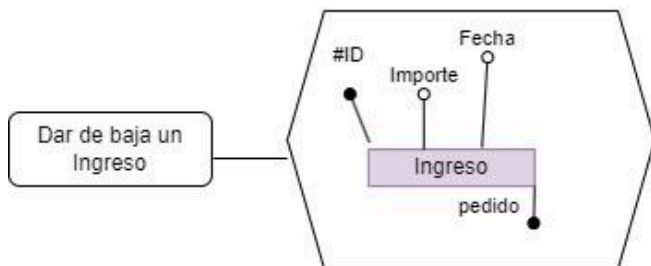
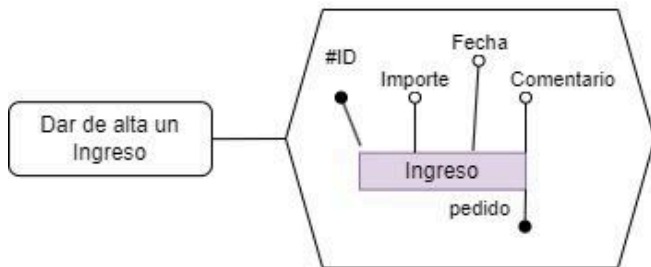
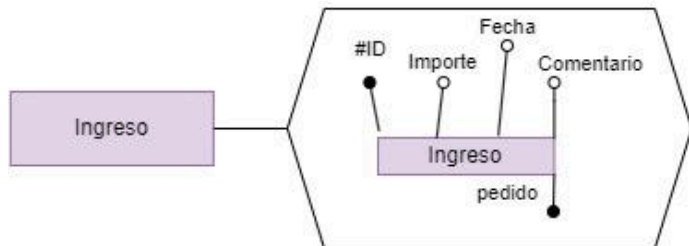
7.1 Human Resources

Person in charge: Thomas Lang



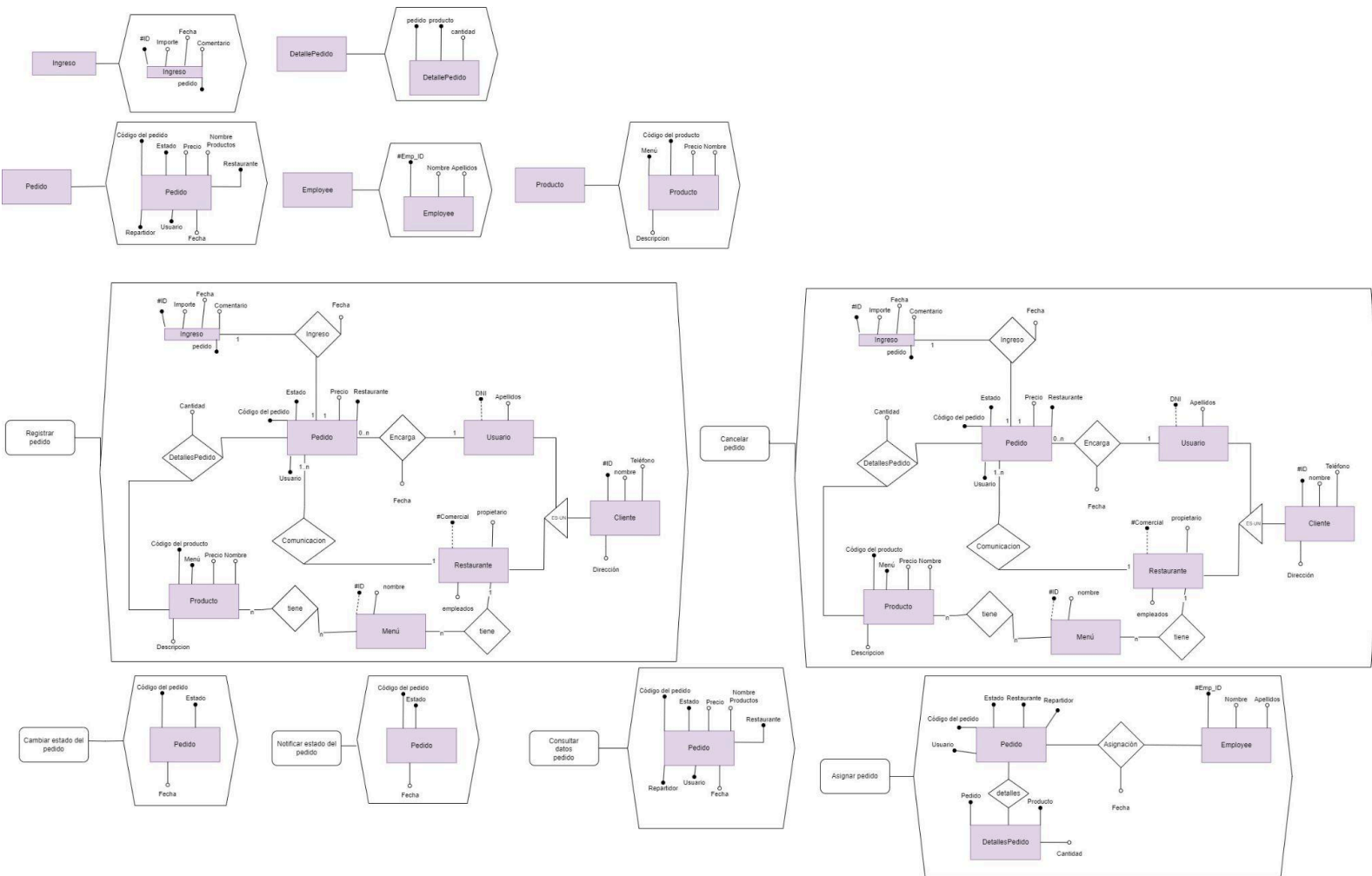
7.2 Contabilidad

Encargado: Alberto Rodríguez Lechuga



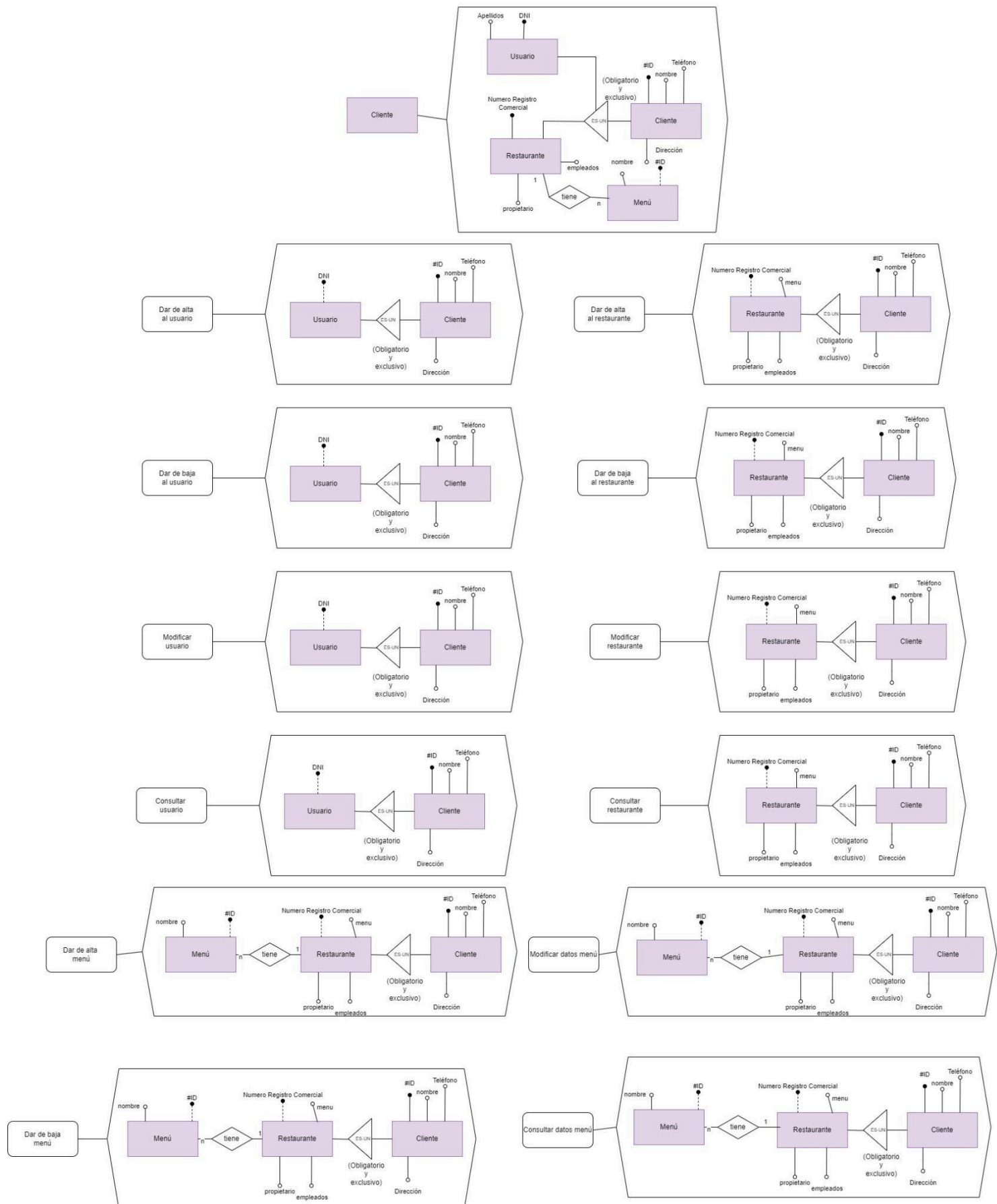
7.3 Logística

Encargado: Mario Illán Valero

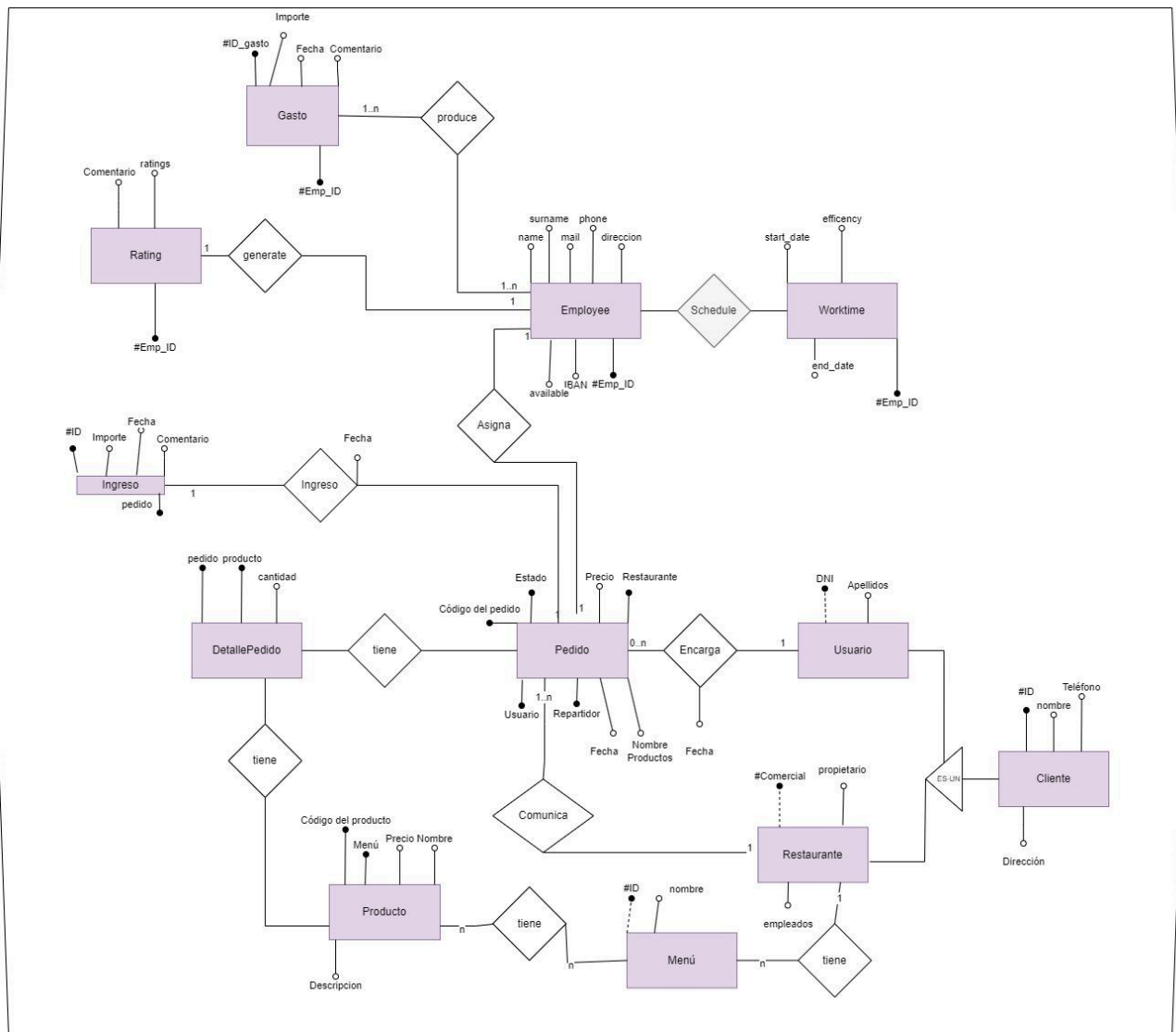


7.4 Cliente

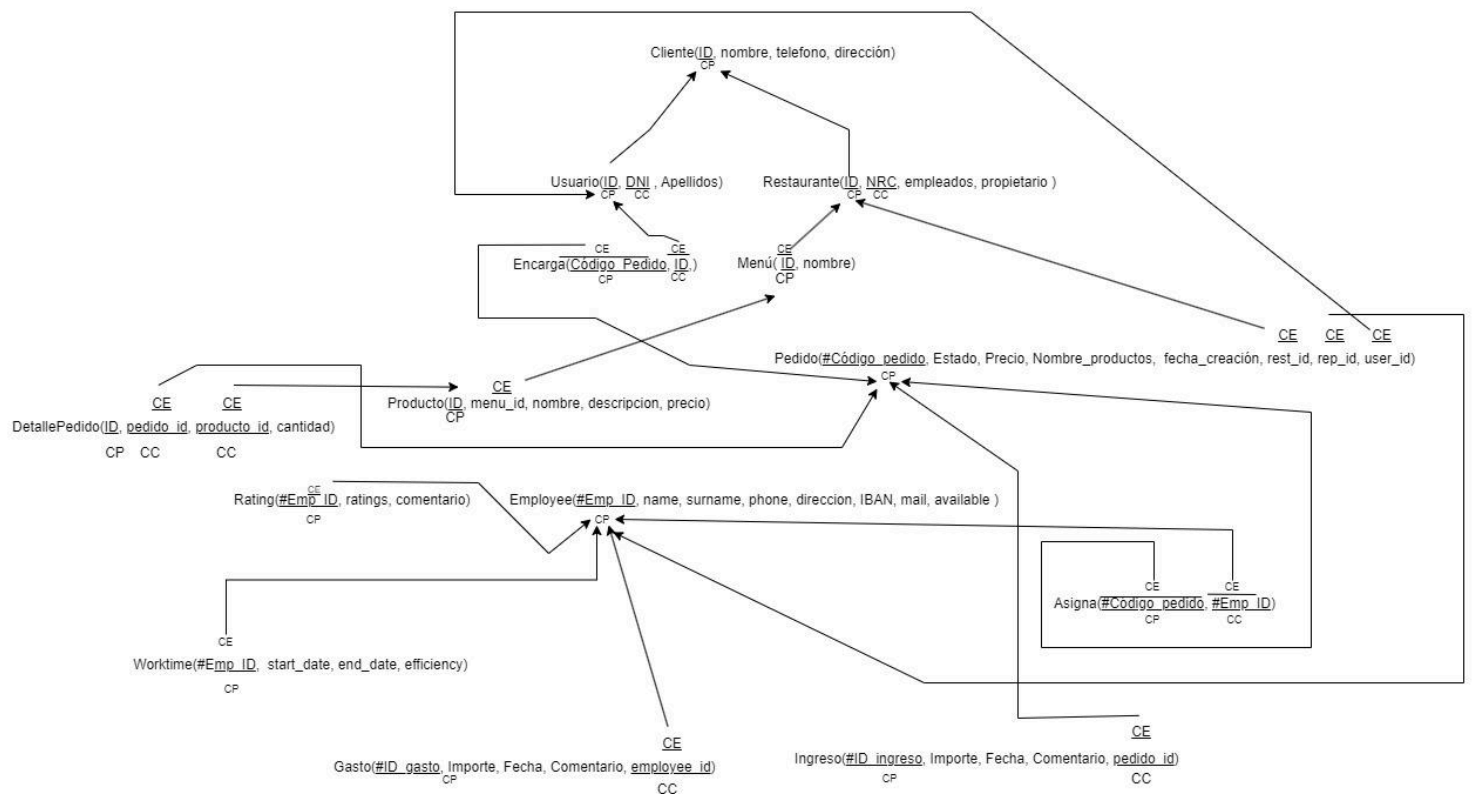
Encargado: Juan Luis Torres Ramos



8. Diagrama E/R



9. Extracción de tablas a partir E/R y Fusión de tablas



10. Justificación de Web Framework y SGBD

Nos vamos a decantar por usar MySQL para nuestro Database Management System y Django para nuestro Web Framework.

MySQL

MySQL es un sistema de gestión de bases de datos relacional (RDBMS) el cual es ampliamente utilizado debido a sus numerosas funciones y características, cuyo código se encuentra en el dominio público y se puede usar tanto de manera privada como comercial. Está desarrollado por Oracle y cuenta con una gran cantidad de aplicaciones.

Una de las características principales, la cual es una gran ventaja, es que permite instalarse en entornos de los principales sistemas operativos, Windows, Mac y Linux.

Otra de sus características es la escalabilidad, ya que tiene soporte para una gran cantidad de registros, tables y filas. A su vez, soporta una gran cantidad de tipos de datos, lo que le aporta una gran versatilidad en cuanto a las situaciones en las que puede implementarse.

Una de las ventajas que ofrece y que ha sido uno de los principales motivos para decantarnos por este SGBD es que cuenta con un amplio soporte para lenguajes de programación, incluyendo múltiples librerías o APIs que permiten integrarlo perfectamente en un sistema desarrollado en cualquier lenguaje.

Por último cabe destacar que cuenta con una documentación oficial actualizada y muy amplia, además de contar con una comunidad enorme para resolver dudas en foros por ejemplo.

Estas son algunas de las características por las que nos hemos decantado por usar MySQL.

Django

Esta herramienta es un framework web de alto nivel y código abierto construido en Python, la cual destaca por ser intuitiva y versátil para el desarrollo estructurado de aplicaciones web. Su flexibilidad nos permite poder escoger nuestro motor para la base de datos sin problemas. También su versatilidad se extiende a la forma de trabajo que nos brinda Django.

Es sencilla de utilizar para los desarrolladores, aparte de ser fácil de instalar y de aprender, ya que muchas de las aplicaciones web que se desarrollan tienen una funcionalidad similar, y esta herramienta nos permite agrupar las funcionalidades en una colección de módulos reutilizables, por lo que puede llegar a ser más eficiente.

Algunas de estas funcionalidades son la autenticación de usuarios o la administración de contenido.

Django tiene una buena comunidad activa, lo que también da un sólido soporte de errores, de actualizaciones, y de documentación. A causa de esta, también tiene un gran soporte en el desarrollo de API , que simplifican y aceleran el trabajo para integrar a nuestro trabajo funcionalidades adicionales . Al estar escrito en Python, tiene una sintaxis simple y eficiente de comprender además de desarrollar.

Por último, Django ofrece varias funciones de seguridad para proteger sus aplicaciones web y a sus usuarios.

Para gestionar nuestro flujo de trabajo, decidimos dividir el grupo para desarrollar por una parte el frontend, y por otro lado el backend

- frontend: Alberto, Thomas
- backend: Juan Luis, Mario

11. Scripts de creación de base de datos e inserción de datos

Tabla 1:

```
CREATE TABLE Cliente (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    Nombre VARCHAR(255),  
    Telefono VARCHAR(9),  
    Direccion VARCHAR(255)  
);
```

```
INSERT INTO Cliente (Nombre, Telefono, Direccion) VALUES  
( 'Juan', '123456789', 'Calle España'),  
( 'Maria', '123123123', 'Calle Valencia'),  
( 'Carlos', '111111111', 'Calle Granada'),  
( 'Restaurante La Plata', '101010101', 'Calle Malaga'),  
( 'Bar Manolo', '123454561', 'Calle Pontevedra'),  
( 'Los Pollos Hermanos', '123123122', 'Calle Barcelona');
```

tabla 2:

```
CREATE TABLE Usuario (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    Apellidos VARCHAR(255) NULL,  
    DNI VARCHAR(9) UNIQUE,  
    FOREIGN KEY (id) REFERENCES Cliente(id)  
);
```

```
INSERT INTO Usuario (Apellidos, DNI, Cliente_id) VALUES  
( 'Gomez', '123456789', 1),  
( 'Martinez', '987654321', 2),  
( 'Rodriguez', '456789012', 3);
```

Tabla 3:

```
CREATE TABLE Restaurante (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    NRC VARCHAR(15) UNIQUE,  
    Empleados INT,  
    Propietario VARCHAR(40),  
    FOREIGN KEY (id) REFERENCES Cliente(id)  
);
```

```
INSERT INTO Restaurante (NRC, Empleados, Propietario, Cliente_id)  
VALUES  
( 'NRC123', 200DC, 'Margarita', 4),  
( 'NRC456', 151AS, 'Manolo', 5),  
( 'NRC789', 251LO, 'Gustavo', 6);
```

Tabla 4:

```
CREATE TABLE Menu (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    restaurante_id INT,  
    FOREIGN KEY (restaurante_id) REFERENCES Restaurante(id) ON  
DELETE CASCADE,  
);
```

```
INSERT INTO Menu (restaurante_id) VALUES  
(4),  
(5),  
(6);
```

el nombre esta en la inicializacion en menu en models.py

Tabla 5:

```
CREATE TABLE Employee (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    Nombre VARCHAR(30),  
    Apellidos VARCHAR(30),  
    Direccion VARCHAR(255),  
    Telefono VARCHAR(9) NULL,  
    IBAN VARCHAR(25) UNIQUE,  
    Mail VARCHAR(30),  
    disponible BOOLEAN DEFAULT true  
);
```

```
INSERT INTO Employee (Nombre, Apellidos, Direccion, Telefono,  
IBAN, Mail, disponible) VALUES  
( 'Paco', 'Miguel', 'Calle Alcalde', '999999999',  
'ES0123456789012345678901', 'paco@email.com', true),  
( 'Juan', 'Juan', 'Calle Bebida', '987654321',  
'ES1122334455667788990011', 'juan@email.com', true),  
( 'Carlos', 'Bautista', 'Calle C', '192192192',  
'ES2233445566778899001122', 'carlos@email.com', true);
```

Tabla 6:

```
CREATE TABLE Rating (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    rating INT,  
    comentario TEXT,  
    empleado_id INT,  
    FOREIGN KEY (empleado_id) REFERENCES Employee(id) ON DELETE  
CASCADE,  
);
```

```
INSERT INTO Rating (rating, comentario, empleado_id) VALUES  
(5, 'Buena atención al cliente', 1),  
(4, 'Comida deliciosa', 2),  
(3, 'Podría mejorar el servicio', 3);
```

Tabla 7:

```
CREATE TABLE Worktime (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    efficiency INT,  
    start_date DATETIME,  
    end_date DATETIME,  
    employee_id INT,  
    FOREIGN KEY (employee_id) REFERENCES Employee(id) ON DELETE  
CASCADE,  
);
```

```
INSERT INTO Worktime (efficiency, start_date, end_date,  
employee_id) VALUES  
(8, '2023-05-01 08:00:00', '2023-05-01 17:00:00', 1),  
(7, '2023-05-02 09:30:00', '2023-05-02 18:30:00', 2),  
(6, '2023-05-03 07:45:00', '2023-05-03 16:45:00', 3);
```

Tabla 8:

```
CREATE TABLE Producto (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    menu_id INT,  
    nombre VARCHAR(100),  
    descripcion TEXT NULL,  
    precio DECIMAL(8, 2),  
    FOREIGN KEY (menu_id) REFERENCES Menu(id) ON DELETE CASCADE,  
);
```

```
INSERT INTO Producto (menu_id, nombre, descripcion, precio) VALUES  
(1, 'Pizza Margarita', 'Deliciosa pizza con tomate y queso  
mozzarella', 10.99),  
(2, 'Ensalada César', 'Ensalada fresca con pollo, lechuga y  
aderezo César', 8.50),  
(3, 'Pasta Bolognesa', 'Pasta con salsa de carne y tomate',  
12.99);
```

Tabla 9:

```
CREATE TABLE Pedido (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    estado VARCHAR(40) DEFAULT 'En Preparación',  
    fecha_creacion DATETIME DEFAULT CURRENT_TIMESTAMP,  
    precio_total DECIMAL(10, 2) DEFAULT 0.0,  
    restaurante_id INT,  
    repartidor_id INT NULL,  
    usuario_id INT NULL,  
    FOREIGN KEY (restaurante_id) REFERENCES Restaurante(id) ON  
DELETE CASCADE,  
    FOREIGN KEY (repartidor_id) REFERENCES Employee(id) ON DELETE  
SET NULL,  
    FOREIGN KEY (usuario_id) REFERENCES Usuario(id) ON DELETE SET  
NULL  
);
```

```
INSERT INTO Pedido (estado, fecha_creacion, precio_total,  
restaurante_id, repartidor_id, usuario_id) VALUES  
( 'En Preparación', CURRENT_TIMESTAMP, 25.99, 1, 1, 1),  
( 'En Proceso', CURRENT_TIMESTAMP, 32.50, 2, 2, 2),  
( 'Entregado', CURRENT_TIMESTAMP, 18.75, 3, 3, 3);
```

Tabla 10

```
CREATE TABLE Asigna (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    trabajador_id INT,  
    pedido_id INT,  
    FOREIGN KEY (trabajador_id) REFERENCES Employee(id) ON DELETE  
CASCADE,  
    FOREIGN KEY (pedido_id) REFERENCES Pedido(id) ON DELETE  
CASCADE,  
    UNIQUE (trabajador_id, pedido_id)  
);
```

```
INSERT INTO Asigna (trabajador_id, pedido_id) VALUES  
(1, 1),  
(2, 2),  
(3, 3);
```

tabla 11:

```
CREATE TABLE DetallePedido (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    pedido_id INT,  
    producto_id INT,  
    cantidad INT DEFAULT 1,  
    FOREIGN KEY (pedido_id) REFERENCES Pedido(id) ON DELETE  
CASCADE,  
    FOREIGN KEY (producto_id) REFERENCES Producto(id) ON DELETE  
CASCADE,  
    UNIQUE (pedido_id, producto_id)  
);
```

```
INSERT INTO DetallePedido (pedido_id, producto_id, cantidad)  
VALUES  
(1, 1, 2),  
(2, 2, 3),  
(3, 3, 1);
```

tabla 12:

```
CREATE TABLE Encarga (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    usuario_id INT,  
    pedido_id INT,  
    FOREIGN KEY (usuario_id) REFERENCES Usuario(id) ON DELETE  
CASCADE,  
    FOREIGN KEY (pedido_id) REFERENCES Pedido(id) ON DELETE  
CASCADE,  
    UNIQUE (usuario_id, pedido_id)  
);
```

```
INSERT INTO Encarga (usuario_id, pedido_id) VALUES  
(1, 1),  
(2, 2),  
(3, 3);
```

tabla 13:

```
CREATE TABLE Ingreso (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    Importe INT,  
    Fecha DATETIME,  
    comentario TEXT NULL,  
    pedido_id INT NULL,  
    FOREIGN KEY (pedido_id) REFERENCES Pedido(id) ON DELETE  
CASCADE,  
    UNIQUE (pedido_id)  
);
```

```
INSERT INTO Ingreso (Importe, Fecha, comentario, pedido_id) VALUES  
(50, '2023-01-01 10:30:00', 'Pago realizado', 1),  
(75, '2023-01-02 12:45:00', 'Compra completa', 2),  
(30, '2023-01-03 09:15:00', NULL, 3);
```

tabla 14:

```
CREATE TABLE Gasto (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    Importe INT,  
    Fecha DATETIME,  
    comentario TEXT NULL,  
    employee_id INT NULL,  
    FOREIGN KEY (employee_id) REFERENCES Employee(id) ON DELETE  
CASCADE  
);
```

```
INSERT INTO Gasto (Importe, Fecha, comentario, employee_id) VALUES  
(20, '2023-01-01 15:00:00', 'Salario de Juan', 1),  
(30, '2023-01-02 11:45:00', 'Salario de Paco', 2),  
(15, '2023-01-03 14:30:00', 'Salario de Carlos', 3);
```


12. Conexión a la base de datos Universidad

Por motivos de eficiencia y recursos, hemos decidido usar la base de datos que implementa Django en su framework, aun así hemos hecho un pequeño script donde nos podemos conectar a la base de datos de oracle de la universidad. La base de datos que usa el servidor es Oracle, Nosotros usamos MySQL.

database_connection.py

```
import cx_Oracle

# 1. Configuración de la conexión
dsn = cx_Oracle.makedsn('oracle0.ugr.es', 1521, service_name='practbd.oracle0.ugr.es')
usuario = 'x0596044'
contraseña = 'x0596044'

# 2. Configuración de la conexión + cadena de conexion
host = 'oracle0.ugr.es'
port = 1521
service_name = 'practbd.oracle0.ugr.es'
usuario = 'x0596044'
contraseña = 'x0596044'
cadena_conexion = f'{usuario}/{contraseña}@{host}:{port}/{service_name}'

try:
    # 3. Conectar a la base de datos
    connection = cx_Oracle.connect(cadena_conexion2)
    print("Conexión exitosa.")

    # 4. Crear un cursor para ejecutar consultas SQL
    cursor = connection.cursor()

    # 5. Con el curso puedo hacer sentencias SQL
    # aquí irían sentencias SQL

except cx_Oracle.Error as e:
    print(f"Error de base de datos: {e}")
finally:
    # 6. Cerrar el cursor y la conexión al finalizar
    if 'cursor' in locals():
        cursor.close()
    if 'connection' in locals():
        connection.close()
    print("Conexión cerrada.")
```

ejecución

```
$ pip install cx_Oracle // tener instalado cx_Oracle
$ python database_connection.py
```

13. Cursos, disparador y excepción

Logística

Evento de MySQL en el que de manera periódica cada minuto se cambia la columna Estado de la tabla Pedido. Cuando se crea Pedido se le asigna de manera DEFAULT el estado 'En Preparación' y al pasar un minuto se cambian los estados de los pedidos que están 'En Preparación' a 'En Envío' y a los que están 'En Envío' a 'Entregado'.

```
1 DELIMITER //
2 • CREATE EVENT cambiar_estado_pedido_evento
3 ON SCHEDULE EVERY 1 MINUTE
4 DO
5 BEGIN
6     UPDATE delivery_pedido
7     SET estado = 'En Envío'
8     WHERE estado = 'En Preparación' AND TIMESTAMPDIFF(MINUTE, created_at, NOW()) >= 1;
9
10    UPDATE delivery_pedido
11    SET estado = 'Entregado'
12    WHERE estado = 'En Envío' AND TIMESTAMPDIFF(MINUTE, created_at, NOW()) >= 1 AND estado != 'Entregado';
13 END;
14 //
15 DELIMITER ;
16
```

Trigger que rellena la tabla Encarga después de insertar en la tabla Pedido un nuevo pedido relacionado con un usuario. Esta tabla nos sirve para controlar los pedidos que encargan los usuarios del subsistema.

```
DELIMITER //

CREATE TRIGGER rellenar_encarga
AFTER INSERT ON delivery_pedido
FOR EACH ROW
BEGIN
    INSERT INTO delivery_encarga (pedido_id, usuario_id)
    VALUES (NEW.id, NEW.usuario_id);
END;

//

DELIMITER ;
```

Trigger que se ejecuta antes de realizar la inserción en la tabla Encarga. Su propósito es verificar si ya existe una entrada en la tabla con la combinación de usuario_id y pedido_id que se está intentando insertar. Si ya existe una coincidencia, el trigger genera una excepción con el mensaje "No puede haber dos usuarios con el mismo pedido", impidiendo la inserción de los datos para que no haya dos usuarios con el mismo pedido.

```
DELIMITER //
```

```
CREATE TRIGGER antes_insertar_encarga  
BEFORE INSERT ON delivery_encarga  
FOR EACH ROW  
BEGIN  
  
    IF EXISTS (SELECT 1 FROM delivery_encarga WHERE usuario_id =  
NEW.usuario_id AND pedido_id = NEW.pedido_id) THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'No puede haber dos usuarios con el mismo  
pedido';  
    END IF;  
END;  
  
//  
  
DELIMITER ;
```

Human Resources

Trigger se activa después de la inserción de un nuevo pedido en la tabla Pedido. Busca un repartidor disponible (disponible = TRUE) y, si lo encuentra, crea una entrada en la tabla Asigna asignando el nuevo pedido al repartidor encontrado. Esto asegura que cada pedido se asigne automáticamente a un repartidor disponible después de su creación.

```
DELIMITER //
```

```
CREATE TRIGGER rellenar_asigna  
AFTER INSERT ON delivery_pedido  
FOR EACH ROW  
BEGIN  
    DECLARE repartidor_disponible_id INT;  
  
    SELECT id INTO repartidor_disponible_id  
    FROM delivery_employee  
    WHERE disponible = TRUE  
    LIMIT 1;  
    IF repartidor_disponible_id IS NOT NULL THEN  
        INSERT INTO delivery_asigna (pedido_id, trabajador_id)  
        VALUES (NEW.id, repartidor_disponible_id);  
    END IF;  
END;  
  
//  
  
DELIMITER ;
```

Trigger que se ejecuta antes de insertar un nuevo registro en la tabla Asigna. Verifica si ya existe una asignación para el mismo trabajador y pedido, y si es así, genera un error para evitar la duplicación de asignaciones.

```
DELIMITER //
```

```
CREATE TRIGGER before_insert_asigna  
BEFORE INSERT ON delivery_asigna  
FOR EACH ROW  
BEGIN  
    IF EXISTS (  
        SELECT 1  
        FROM delivery_asigna  
        WHERE trabajador_id = NEW.trabajador_id and pedido_id =  
new.pedido_id  
    ) THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Ya existe una asignación para este  
trabajador y pedido';  
    END IF;  
END;
```

```
//
```

```
DELIMITER ;
```

Contabilidad

Trigger que se ejecuta antes de la inserción en la tabla Ingreso y evita que se registren ingresos con importes negativos, generando una excepción si se intenta hacer.

```
DELIMITER //
```

```
CREATE TRIGGER evitar_ingreso_negativo  
BEFORE INSERT ON delivery_ingreso  
FOR EACH ROW  
BEGIN  
    IF NEW.Importe < 0 THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'No se permiten ingresos negativos';  
    END IF;  
END;  
  
//  
  
DELIMITER ;
```

Trigger que se ejecuta antes de la inserción en la tabla Gasto y evita que se registren gastos con montos negativos, generando una excepción si se intenta hacer.

```
DELIMITER //
```

```
CREATE TRIGGER evitar_gasto_negativo  
BEFORE INSERT ON delivery_gasto  
FOR EACH ROW  
BEGIN  
    IF NEW.monto < 0 THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'No se permiten gastos negativos';  
    END IF;  
END;  
  
//  
  
DELIMITER ;
```

Cliente

Trigger que se ejecuta antes de la inserción en la tabla Usuario y evita que se ingresen usuarios con el mismo DNI.

```
DELIMITER //
```

```
CREATE TRIGGER evitar_duplicados_dni  
BEFORE INSERT ON delivery_usuario  
FOR EACH ROW  
BEGIN  
    IF EXISTS (SELECT 1 FROM delivery_usuario WHERE DNI = NEW.DNI) THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Ya existe un usuario con este DNI';  
    END IF;  
END;  
  
//  
  
DELIMITER ;
```

Trigger que se ejecuta antes de la inserción en la tabla Restaurante y evita que se ingresen restaurantes con el mismo NRC.

```
DELIMITER //
```

```
CREATE TRIGGER evitar_duplicados_nrc  
BEFORE INSERT ON delivery_restaurante  
FOR EACH ROW  
BEGIN  
    IF EXISTS (SELECT 1 FROM delivery_restaurante WHERE NRC = NEW.NRC)  
    THEN  
        SIGNAL SQLSTATE '45000'  
        SET MESSAGE_TEXT = 'Ya existe un restaurante con este NRC';  
    END IF;  
END;  
  
//  
  
DELIMITER ;
```