



UNIVERSIDAD
DE GRANADA

PROYECTO FINAL

Diseño y Desarrollo de Sistemas de Información

AUTORES:

ALONSO BUENO HERRERO
MANUEL CASTELLÓN REGUERO
PABLO LOMBARDEO ROS
FRANCISCO JAVIER TORRES BAREA

Grado en Ingeniería Informática

Tercer curso

Curso 2019-20

Última actualización: 17/03/20

Nota preliminar

Presentamos aquí toda la documentación asociada al proyecto de prácticas de la asignatura. Se incluyen los apartados indicados en la entrega final: desde el análisis de requisitos hasta el código asociado a los disparadores y la implementación en Java de cuatro requisitos funcionales.

En Granada, a 17 de marzo de 2020.

Contenido

Nota preliminar.....	1
1 Requisitos funcionales.....	3
1.1 Subsistema 1	3
1.2 Subsistema 2	3
1.3 Subsistema 3	4
1.4 Subsistema 4	5
2 Requisitos de datos.....	6
3 Restricciones semánticas.....	12
3.1 Subsistema 1	12
3.2 Subsistema 2	13
3.3 Subsistema 3	13
3.4 Subsistema 4	13
4 Validación cruzada de requisitos	14
4.1 Validación cruzada de requisitos funcionales.....	14
4.2 Validación cruzada de requisitos de datos.....	14
4.3 Validación cruzada de restricciones semánticas	15
5 Diagramas de flujo de datos y esquemas externos.....	16
5.1 Paso a tablas.....	36
6 Implementación.....	39
6.1 Diseño físico de la base de datos: sentencias de creación de tablas	39
6.2 Inserción de datos de prueba	42
6.3 Disparadores.....	45
6.3.1 Subsistema 1	45
6.3.2 Subsistema 2	46
6.3.3 Subsistema 3	46
6.3.4 Subsistema 4	47
6.4 Interfaces implementadas	48
6.4.1 Subsistema 1	48
6.4.2 Subsistema 2	49
6.4.3 Subsistema 3	50
6.4.4 Subsistema 4	52

1 Requisitos funcionales

1.1 Subsistema 1

RF	1
Nombre	Añadir entidad
Descripción	Añadir una nueva entidad
Entrada	RD1
Información almacenada	RD2
Salida	

RF	2
Nombre	Consultar colaboradores en una edición
Descripción	Permite consultar los colaboradores que participan en una determinada edición.
Entrada	RD3
Información almacenada	RD4
Salida	RD5

RF	3
Nombre	Registrar patrocinador
Descripción	Añadir un nuevo patrocinador
Entrada	RD6
Información almacenada	RD7
Salida	

RF	4
Nombre	Consultar dinero total aportado por colaboradores de una edición
Descripción	Muestra el dinero total aportado entre todos los colaboradores de la edición indicada
Entrada	RD8
Información almacenada	RD9
Salida	RD10

1.2 Subsistema 2

RF	5
Nombre	Añadir pista
Descripción	Añadir una nueva pista al sistema
Entrada	RD11
Información almacenada	RD12
Salida	

RF	6
Nombre	Insertar partido
Descripción	Añadir un nuevo partido al torneo
Entrada	RD13
Información almacenada	RD14
Salida	

RF	7
Nombre	Añadir resultado a un partido
Descripción	Establecer el resultado de un partido que ha terminado.
Entrada	RD15
Información almacenada	RD16
Salida	

RF	8
Nombre	Mostrar partidos en una fecha
Descripción	Mostrar datos de partidos que se juegan en una fecha concreta.
Entrada	RD17
Información almacenada	RD18
Salida	RD19

1.3 Subsistema 3

RF	9
Nombre	Añadir material
Descripción	Añadir un nuevo material al sistema
Entrada	RD20
Información almacenada	RD21
Salida	

RF	10
Nombre	Realizar pedido
Descripción	Realizar un pedido de materiales en un pista
Entrada	RD22
Información almacenada	RD23
Salida	RD39

RF	11
Nombre	Entregar pedido
Descripción	Registra el trabajador y la fecha a la que se recoge el pedido
Entrada	RD24

Información almacenada	RD25
Salida	RD40

RF	12
Nombre	Consultar trabajadores que tienen que recoger un pedido en una fecha
Descripción	Consulta los trabajadores que tiene que recoger pedidos y llevarlos a una pista en una fecha dada
Entrada	RD26
Información almacenada	RD27
Salida	RD28

1.4 Subsistema 4

RF	13
Nombre	Trabajar en una edición
Descripción	Un trabajador realiza un trabajo en una determinada edición
Entrada	RD29
Información almacenada	RD30
Salida	

RF	14
Nombre	Asignar horario de un trabajo
Descripción	Se realiza la asignación del horario de trabajo a un trabajador
Entrada	RD31
Información almacenada	RD32
Salida	RD41

RF	15
Nombre	Generar listado de trabajadores libres
Descripción	Devuelve los nombres de los trabajadores disponibles para realizar un trabajo
Entrada	RD33
Información almacenada	RD34
Salida	RD35

RF	16
Nombre	Mostrar horario de trabajador
Descripción	Devuelve el horario de un trabajador en una fecha concreta
Entrada	RD36
Información almacenada	RD37
Salida	RD38

2 Requisitos de datos

RD	1
Nombre	Datos de nueva entidad
Descripción	Datos que se utilizan a la hora de registrar una nueva entidad en el sistema.
Contenido	<ul style="list-style-type: none"> - nombre (cadena de 40 caracteres), - teléfono (cadena de 12 caracteres, todos numéricos menos el primero, que puede ser un +). - correo electrónico (cadena de 50 caracteres que ha de contener el carácter @). - nombre de persona de contacto (cadena de hasta 100 caracteres).

RD	2
Nombre	Datos de entidad almacenada
Descripción	Datos que se utilizan a la hora de almacenar una nueva entidad en el sistema.
Contenido	<ul style="list-style-type: none"> - nombre (cadena de 40 caracteres), - teléfono (cadena de 12 caracteres, todos numéricos menos el primero, que puede ser un +). - correo electrónico (cadena de 50 caracteres que ha de contener el carácter @). - nombre de persona de contacto (cadena de hasta 100 caracteres).

RD	3
Nombre	Número de edición a consultar
Descripción	Número de la edición de la que se quieren consultar los colaboradores
Contenido	<ul style="list-style-type: none"> - Número de la edición (número entero de 3 cifras)

RD	4
Nombre	Datos de colaboradores almacenados de una edición
Descripción	Datos de cada uno de los colaboradores del sistema registrados para el número de edición buscado.
Contenido	<ul style="list-style-type: none"> - Número de la edición (número entero de 3 cifras) - Colaboradores

RD	5
Nombre	Datos de los colaboradores registrados en la edición indicada
Descripción	Datos de cada uno de los colaboradores registrados en la edición consultada
Contenido	<ul style="list-style-type: none"> - nombre (cadena de 40 caracteres), - teléfono (cadena de 12 caracteres, todos numéricos menos el primero, que puede ser un +). - correo electrónico (cadena de 50 caracteres que ha de contener el carácter @). - nombre de persona de contacto (cadena de hasta 100 caracteres).

RD	6
Nombre	Datos de nuevo patrocinador

Descripción	Datos necesarios para registrar una entidad del sistema como patrocinadora
Contenido	<ul style="list-style-type: none"> - Patrocinador - Número de edición (número entero de 3 cifras) - Dinero con el que colabora el patrocinador (número decimal de 6 cifras enteras y 2 decimales)

RD	7
Nombre	Datos de patrocinador almacenados
Descripción	Datos necesarios para almacenar un patrocinador
Contenido	<ul style="list-style-type: none"> - Patrocinador - Número de entidad (número entero de 3 cifras) - Dinero con el que colabora el patrocinador (número decimal de 6 cifras enteras y 2 decimales)

RD	8
Nombre	Edición consultada
Descripción	Número de la edición de la que se quiere consultar el dinero total aportado por los colaboradores de la misma
Contenido	<ul style="list-style-type: none"> - Número de la edición (número entero de 3 cifras)

RD	9
Nombre	Dinero total aportado por los colaboradores de una edición.
Descripción	Suma del dinero aportado por todos y cada uno de los colaboradores de la edición consultada
Contenido	<ul style="list-style-type: none"> - dinero total aportado (número decimal de 8 cifras enteras y 2 decimales).

RD	10
Nombre	Dinero total aportado por los colaboradores de una edición.
Descripción	Suma del dinero aportado por todos y cada uno de los colaboradores de la edición consultada
Contenido	<ul style="list-style-type: none"> - dinero total aportado (número decimal de 8 cifras enteras y 2 decimales).

RD	11
Nombre	Datos de nueva pista
Descripción	Datos de la pista que se quiere añadir.
Contenido	<ul style="list-style-type: none"> - nombre: cadena de 30 caracteres; - capacidad: número entero de 4 cifras

RD	12
Nombre	Datos de la pista almacenados
Descripción	Datos de la pista que se almacenan.
Contenido	<ul style="list-style-type: none"> - nombre: cadena 30 caracteres. - capacidad: número entero de 4 cifras

RD	13
-----------	----

Nombre	Datos del nuevo partido
Descripción	Datos del partido que se quiere añadir.
Contenido	<ul style="list-style-type: none"> - Datos de jugadores - nombre de la pista: cadena de 30 caracteres - árbitro - fecha: dd/mm/aa + hh/mm

RD	14
Nombre	Datos del partido almacenados
Descripción	Datos del partido que se almacenan.
Contenido	<ul style="list-style-type: none"> - Datos de jugadores - nombre de la pista: cadena de 30 caracteres - árbitro - fecha: dd/mm/aa + hh/mm

RD	15
Nombre	Nuevo resultado de un partido
Descripción	Resultado del partido que se quiere añadir
Contenido	<ul style="list-style-type: none"> - valor entero con el resultado del local - valor entero con el resultado del visitante - datos del partido

RD	16
Nombre	Resultado almacenado de un partido
Descripción	Resultado del partido para cada uno de los equipos
Contenido	<ul style="list-style-type: none"> - valor entero con el resultado del local - valor entero con el resultado del visitante - datos del partido

RD	17
Nombre	Fecha de los partidos a consultar
Descripción	Fecha proporcionada para consultar los partidos que se juegan en dicha fecha.
Contenido	<ul style="list-style-type: none"> - fecha (dd/mm/aa+hh/mm)

RD	18
Nombre	Partidos consultados en la fecha
Descripción	Conjunto de partidos consultados
Contenido	<ul style="list-style-type: none"> - datos de los partidos

RD	19
Nombre	Partidos asociados a una fecha
Descripción	Lista de partidos que se juegan en una fecha especificada.
Contenido	<ul style="list-style-type: none"> - jugadores - pista - resultado del partido (si lo hay)

RD	20
Nombre	Datos de entrada de materiales
Descripción	Datos de entrada para cada uno de los materiales que se van a añadir al sistema
Contenido	<ul style="list-style-type: none"> - Nombre del material (cadena de caracteres de 30 caracteres) - Cantidad del material (número entero de 5 cifras) - Datos del patrocinador

RD	21
Nombre	Datos almacenados de materiales
Descripción	Datos que almacenamos de cada uno de los distintos materiales que se van a añadir al sistema
Contenido	<ul style="list-style-type: none"> - Nombre del material (cadena de caracteres de 30 caracteres) - Cantidad del material (número entero de 5 cifras) - Datos del patrocinador

RD	22
Nombre	Datos de entrada de los pedidos
Descripción	Datos del pedido que se va a realizar
Contenido	<ul style="list-style-type: none"> - Nombre de materiales para la pista (lista de cadena de caracteres de 30 caracteres) - Datos de pedido - Cantidad necesitada de cada uno de los materiales (número entero de 5 cifras) - Nombre de la pista en la que se entrega el material (cadena de 30 caracteres)

RD	23
Nombre	Datos de almacenados de los pedidos
Descripción	Datos utilizados para registrar un pedido en el sistema
Contenido	<ul style="list-style-type: none"> - Nombre de materiales para la pista (lista de cadena de caracteres de 30 caracteres) - Datos de pedido - Cantidad necesitada de cada uno de los materiales (número entero de 5 cifras) - Nombre de la pista en la que se entrega el material (cadena de 30 caracteres)

RD	24
Nombre	Datos de entrada de entregar pedido
Descripción	Se proporcionarán los datos necesarios para que el trabajador pueda entregar el pedido en la pista correspondiente
Contenido	<ul style="list-style-type: none"> - Datos de pedido

RD	25
Nombre	Datos almacenados de entrega de un pedido

Descripción	Se consultará el número de pedido para confirmar si un pedido ha sido entregado o no
Contenido	- Datos de pedido

RD	26
Nombre	Datos de pedido y fecha
Descripción	Se especificará la fecha en la cual se quiere consultar los trabajadores que tienen que recoger un pedido
Contenido	- Fecha

RD	27
Nombre	Datos de trabajadores que recogen ese pedido en la fecha dada
Descripción	Datos de trabajadores que recogen ese pedido en la fecha dada
Contenido	- Lista de trabajadores que tienen que recoger algún pedido

RD	28
Nombre	Datos de salida de consultar trabajadores que tienen que recoger un pedido en una fecha
Descripción	Se especificará una lista con todos los trabajadores que tienen que recoger algún pedido en la fecha especificada
Contenido	- Lista de trabajadores que tienen que recoger un pedido

RD	29
Nombre	Datos del trabajador de entrada
Descripción	Datos que permiten identificar a un trabajador
Contenido	<ul style="list-style-type: none"> - Nombre (cadena 30 caracteres) - Apellidos (cadena 60 caracteres) - Correo electrónico (cadena de 50 caracteres) - Teléfono (cadena de 12 caracteres) - Número edición (número entero de 4 dígitos)

RD	30
Nombre	Datos del trabajador almacenados
Descripción	Datos que se almacenan sobre un trabajador que trabaja en una edición
Contenido	<ul style="list-style-type: none"> - Nombre (cadena 30 caracteres) - Apellidos (cadena 60 caracteres) - Correo electrónico (cadena de 50 caracteres) - Teléfono (cadena de 12 caracteres) - Número edición (número entero de 4 dígitos)

RD	31
Nombre	Datos de horario de trabajo de entrada
Descripción	Datos del horario de un trabajo concreto para un determinado trabajador
Contenido	<ul style="list-style-type: none"> - Datos trabajador - Fecha (dd/mm/aa) - Hora inicio (hh:mm)

	<ul style="list-style-type: none"> - Hora fin (hh:mm) - Número pista (número entero de 2 dígitos)
--	---

RD	32
Nombre	Datos de horario de trabajo almacenados
Descripción	Datos del horario de un trabajo concreto para un determinado trabajador
Contenido	<ul style="list-style-type: none"> - Datos trabajador - Fecha (dd/mm/aa) - Hora inicio (hh:mm) - Hora fin (hh:mm) - Número pista (número entero de 2 dígitos)

RD	33
Nombre	Datos de hora de entrada
Descripción	Información sobre una hora concreta de una fecha en una edición
Contenido	<ul style="list-style-type: none"> - Número de edición (número entero de 4 dígitos) - Fecha (dd/mm/aa) - Hora (hh:mm)

RD	34
Nombre	Datos del trabajador almacenados
Descripción	Datos que permiten identificar a un trabajador
Contenido	<ul style="list-style-type: none"> - Nombre (cadena 30 caracteres) - Apellidos (cadena 60 caracteres) - Correo electrónico (cadena de 50 caracteres con @) - Teléfono (cadena de 12 caracteres, primer carácter puede ser +) - Horario - Número edición (número entero de 4 dígitos)

RD	35
Nombre	Datos del trabajador libre de salida
Descripción	Datos que permiten identificar a un trabajador libre
Contenido	<ul style="list-style-type: none"> - Nombre (cadena 30 caracteres) - Apellidos (cadena 60 caracteres) - Correo electrónico (cadena de 50 caracteres con @) - Teléfono (cadena de 12 caracteres, primer carácter puede ser +)

RD	36
Nombre	Fecha de consulta de entrada
Descripción	Fecha cuyo horario se quiere consultar
Contenido	<ul style="list-style-type: none"> - Datos de trabajador - Fecha (dd/mm/aa) - Número edición (número entero de 4 dígitos)

RD	37
Nombre	Datos de horario de un trabajador

Descripción	Datos del horario de un determinado trabajador en una fecha concreta
Contenido	<ul style="list-style-type: none"> - Fecha (dd/mm/aa) - Horas - Pista - Número edición

RD	38
Nombre	Datos de horario salida
Descripción	Diferentes trabajos de un determinado trabajador a lo largo de un día
Contenido	<ul style="list-style-type: none"> - Fecha (dd/mm/aa) - Horas - Pista - Número edición

RD	39
Nombre	Datos almacenados de los pedidos
Descripción	Datos utilizados para registrar un pedido en el sistema
Contenido	<ul style="list-style-type: none"> - Nombre de materiales para la pista (lista de cadena de caracteres de 30 caracteres) - Datos de pedido (número entero de 4 dígitos) - Cantidad necesitada de cada uno de los materiales (número entero de 5 cifras) - Nombre de la pista en la que se entrega el material (cadena de 30 caracteres)

RD	40
Nombre	Datos de pedido entregado
Descripción	Datos del pedido que se van a enviar al trabajador.
Contenido	<ul style="list-style-type: none"> - Nombre del trabajador - DNI del trabajador - Datos de pedido - Fecha de la entrega

RD	41
Nombre	Datos de horario de salida
Descripción	Diferentes trabajos de un determinado trabajador a lo largo de un día
Contenido	<ul style="list-style-type: none"> - Fecha (dd/mm/aa) - Horas - Pista - Datos de edición

3 Restricciones semánticas

3.1 Subsistema 1

RS	1
Nombre	Entidad como colaboradora y patrocinadora
Descripción	No se puede registrar como patrocinadora una entidad previamente registrada como colaboradora y viceversa
RF	3

RDs	RD6, RD7
------------	----------

3.2 Subsistema 2

RS	2
Nombre	Número máximo de partidos diarios en una pista
Descripción	No se pueden jugar más de tres partidos al día en una misma pista.
RF	RF7 (Insertar partido)
RDs	RD13, RD14

RS	3
Nombre	Mínimo número de horas entre partidos
Descripción	Dentro de una jornada, cada partido debe separarse del siguiente/anterior al menos 3 horas
RF	RF7
RDs	RD13, RD14

3.3 Subsistema 3

RS	4
Nombre	Realizar pedidos del mismo patrocinador
Descripción	Un pedido sólo puede contener materiales de un mismo patrocinador
RF	RF9
RDs	RD22

RS	5
Nombre	Diferencia de tiempo entre pedidos asignados a un trabajador
Descripción	Un trabajador no puede tener asignado dos pedidos con menos de una hora de diferencia
RF	RF11
RDs	RD24, RD25

3.4 Subsistema 4

RS	6
Nombre	Horarios solapados
Descripción	Un trabajador debe descansar 1 hora entre dos turnos de trabajo
RF	14
RDs	RD31, RD32

RS	7
Nombre	Misma ubicación
Descripción	Un trabajador no puede trabajar en la misma ubicación en el mismo día
RF	14
RDs	RD31, RD32

4 Validación cruzada de requisitos

4.1 Validación cruzada de requisitos funcionales

RF	Entrada (RD)	Almacenamiento (RD)	Salida (RD)
1	RD1	RD2	
2	RD3	RD4	RD5
3	RD6	RD7	
4	RD8	RD9	RD10
5	RD11	RD12	
6	RD13	RD14	
7	RD15	RD16	
8	RD17	RD18	RD19
9	RD20	RD21	
10	RD22	RD23	RD39
11	RD24	RD25	RD40
12	RD26	RD27	RD28
13	RD29	RD30	
14	RD31	RD32	RD41
15	RD33	RD34	RD35
16	RD36	RD37	RD38

4.2 Validación cruzada de requisitos de datos

RD	Entrada (RF)	Almacenamiento (RF)	Salida (RF)
	RF1		
		RF1	
	RF2		
		RF2	
			RF2
	RF3		
		RF3	
	RF4		
		RF4	
			RF4
	RF5		
		RF5	
	RF6		
		RF6	
	RF7		
		RF7	
	RF8		
		RF8	
			RF8

	RF9		
		RF9	
	RF10		
		RF10	
	RF11		
		RF11	
	RF12		
		RF12	
			RF12
	RF13		
		RF13	
	RF14		
		RF14	
	RF15		
		RF15	
			RF15
	RF16		
		RF16	
			RF16
1			RF10
2			RF11
3			RF14

4.3 Validación cruzada de restricciones semánticas

RS	RF	RDs
1	3	6,7
2	7	13,14
3	7	13,14
4	9	20
5	11	24
6	14	32
7	14	32

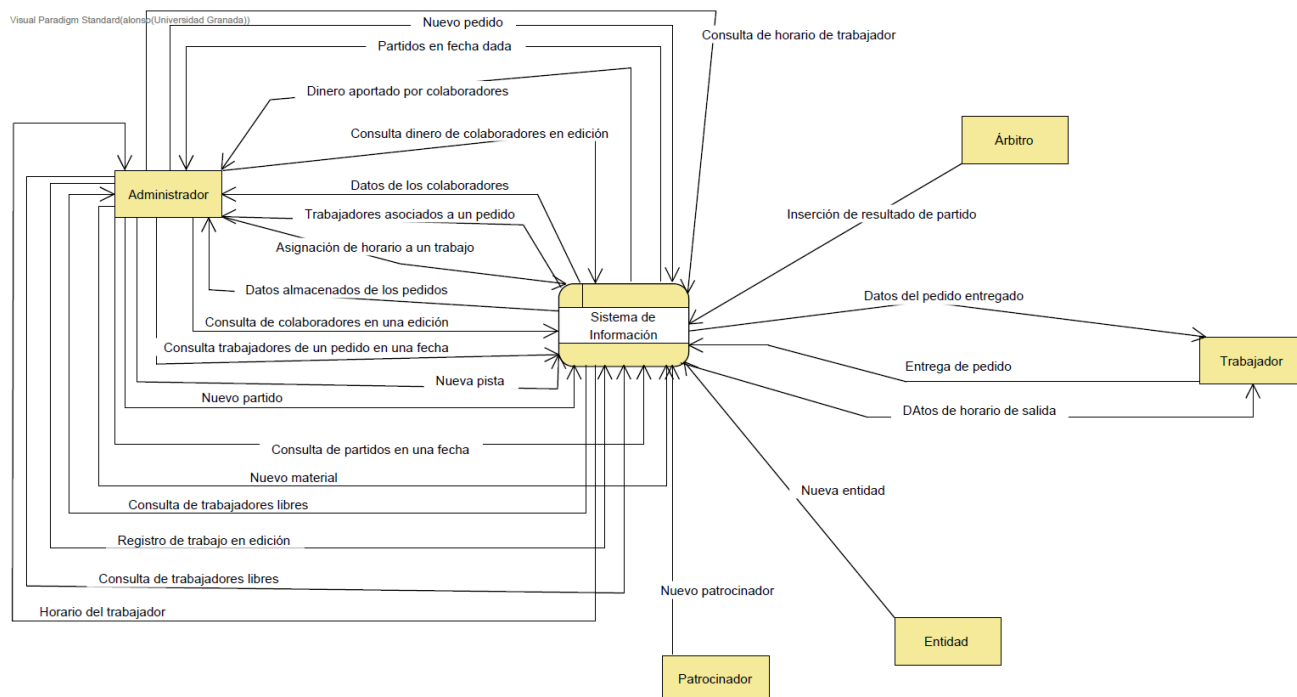
5 Diagramas de flujo de datos y esquemas externos

El contenido de esta sección es (extraído de PRADO) el que se muestra a continuación, por orden de aparición:

1. Esquema de caja negra.
2. Esquema almacén F (DFD0), esquemas externos del DFD0 y esquema almacén D(ER0) como unión de los esquemas externos.
3. Refinamientos (empezando desde $n = 0$):
 - a. Refinamientos parciales de los elementos del DFD n que lo requieran.
 - b. La unión de dichos refinamientos parciales en un único DFD $n+1$.
 - c. Los esquemas externos de los elementos del DFD $n+1$.
 - d. La unión de dichos esquemas externos en un único ER $n+1$.
 - e. La repetición de los pasos anteriores para cada elemento refinable del DFD resultante generando nuevos niveles de refinamiento (DFD1, DFD2, ...) hasta que se determine que el DFD resultante no requiere refinamientos posteriores ni en procesos ni en almacenes.
4. Esquema ER final con todos sus atributos y cardinalidades.

5.1 Fase de diseño: diagramas DFD y esquemas externos

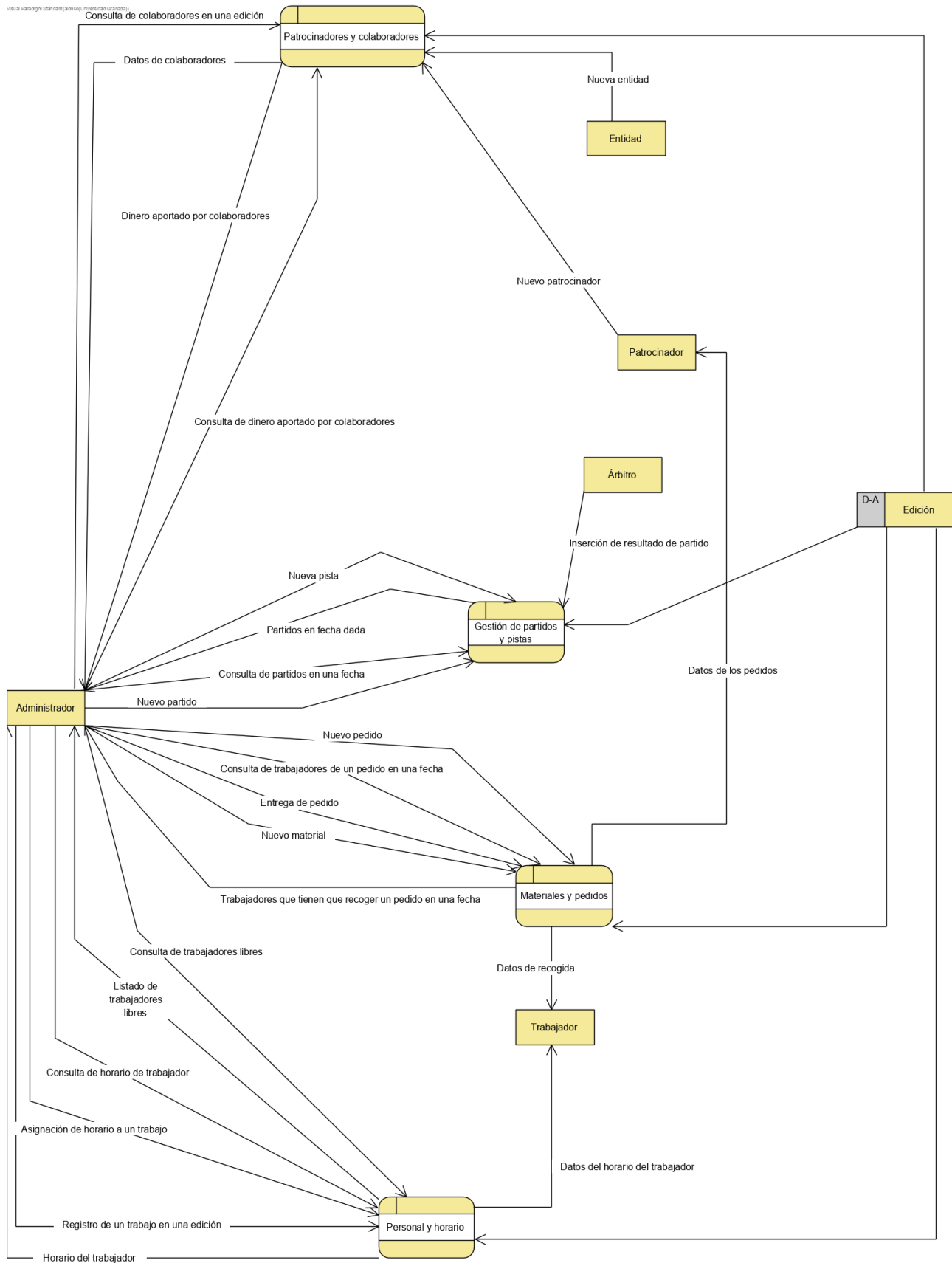
1.- Esquema de caja negra



2.- Esquema almacén F (DFD0), esquemas externos del DFD0 y esquema almacén D(ER0) como unión de los esquemas externos.

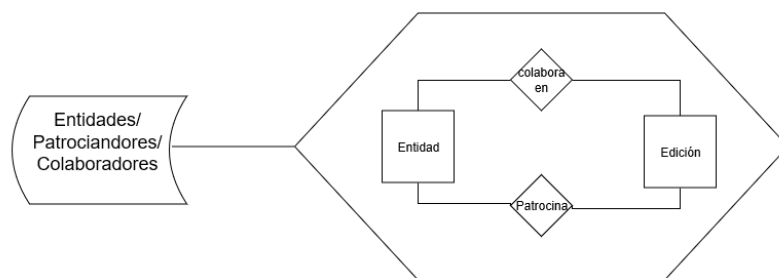
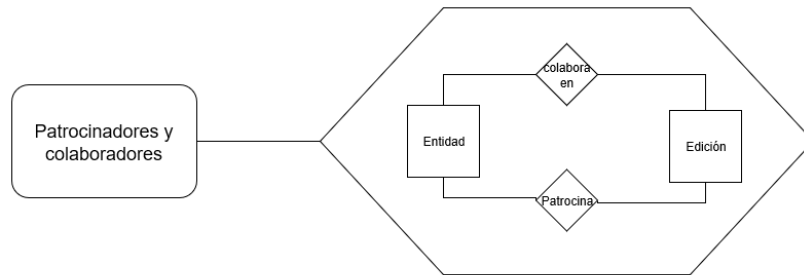
(ver página siguiente)

2.1.- DFD0

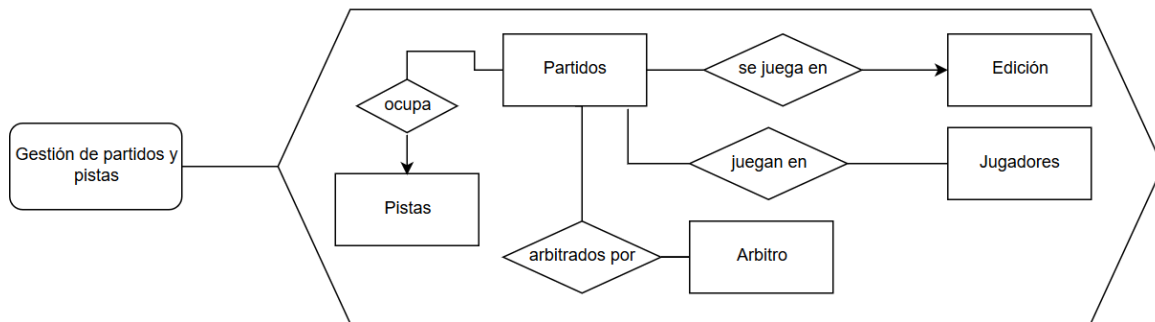


2.2.- Esquemas externos para el DFD0 (uno por cada proceso/subsistema)

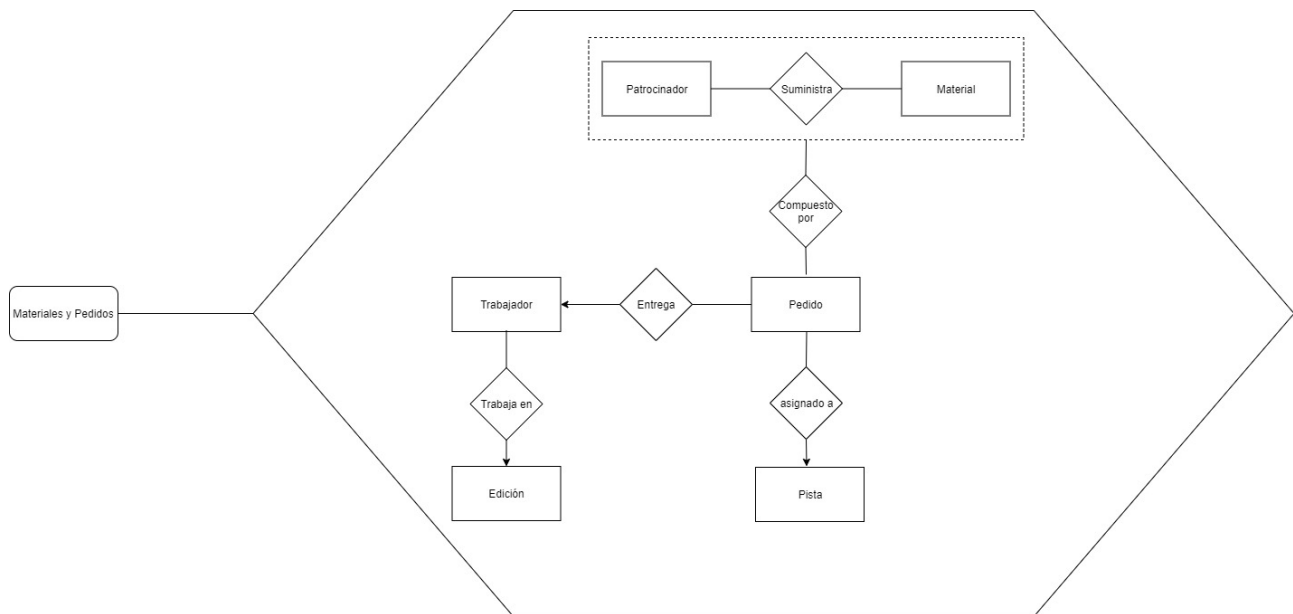
Proceso 1.-



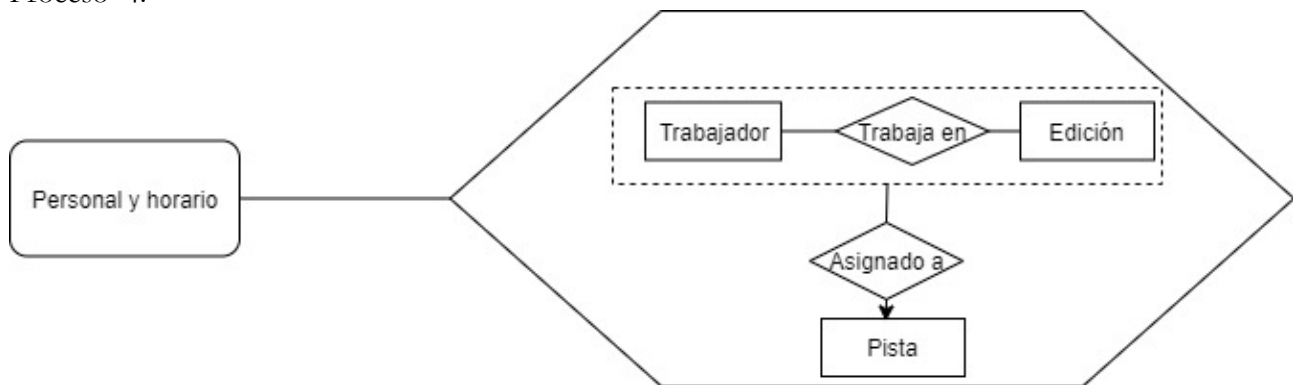
Proceso 2.-



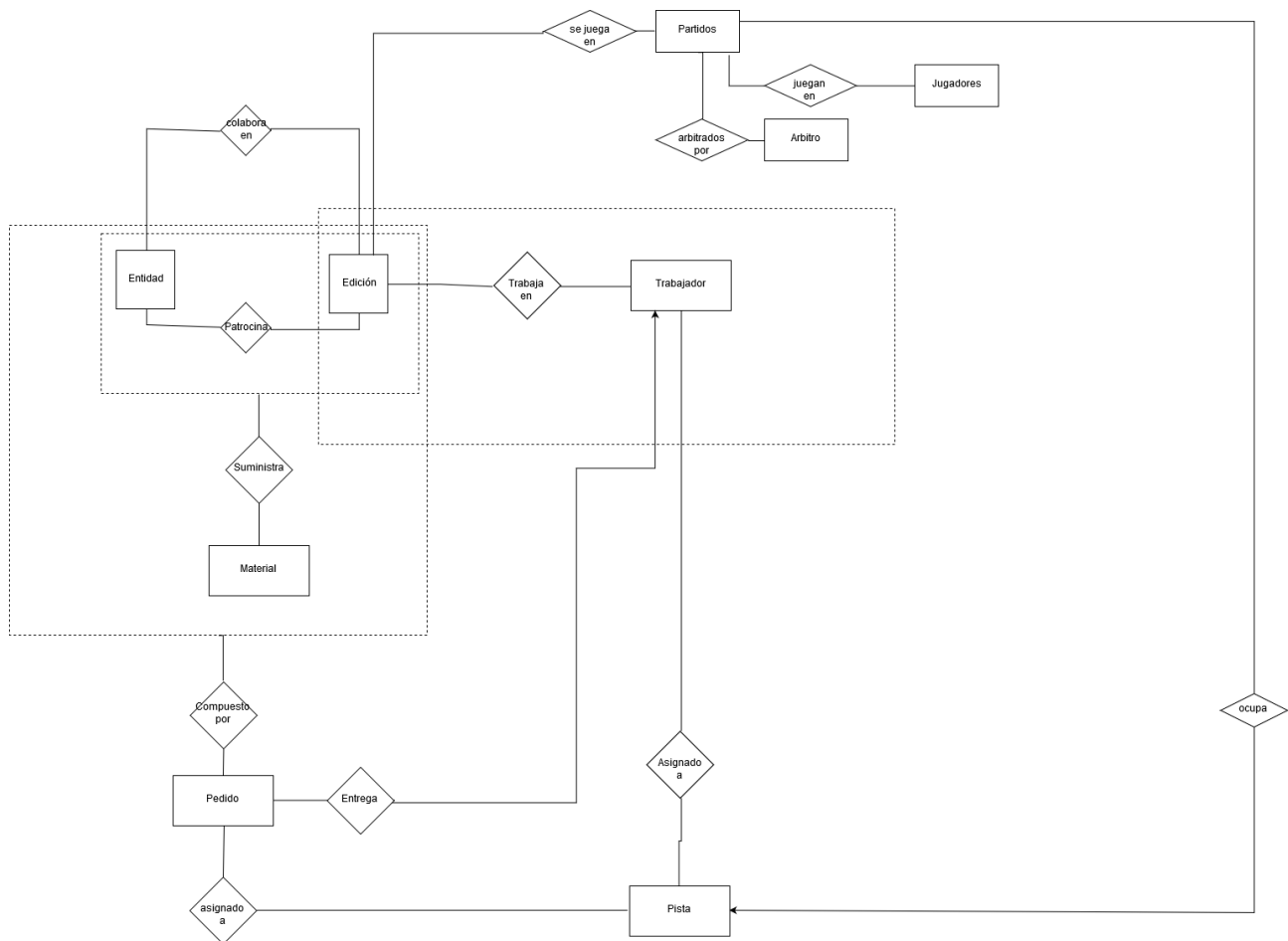
Proceso 3.-



Proceso 4.-



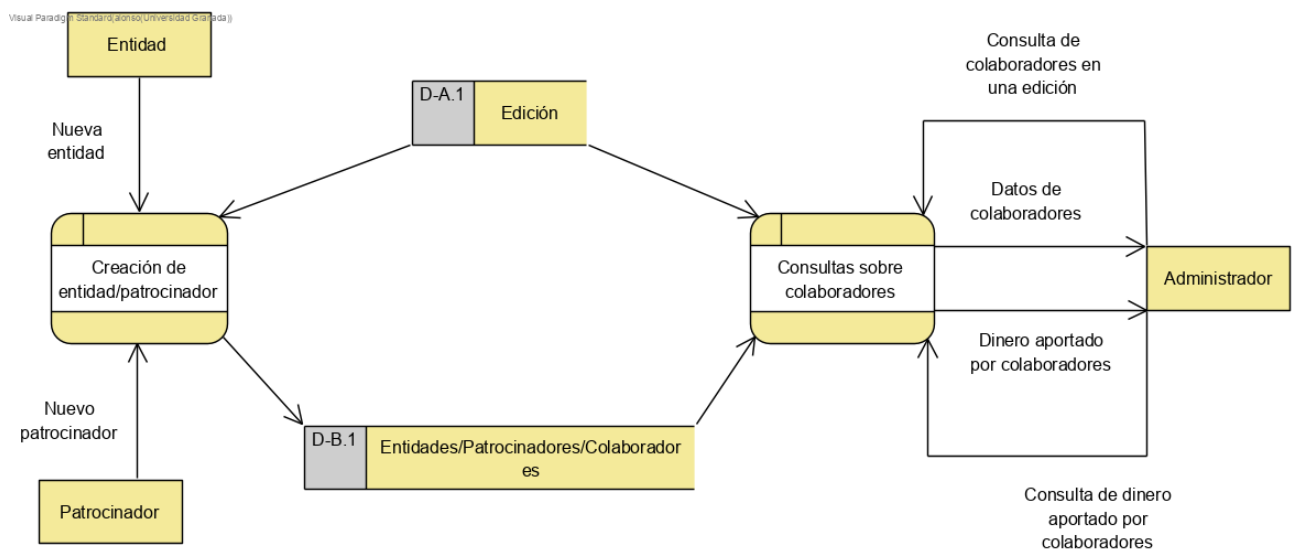
2.3.- Esquema almacén D(ER0)



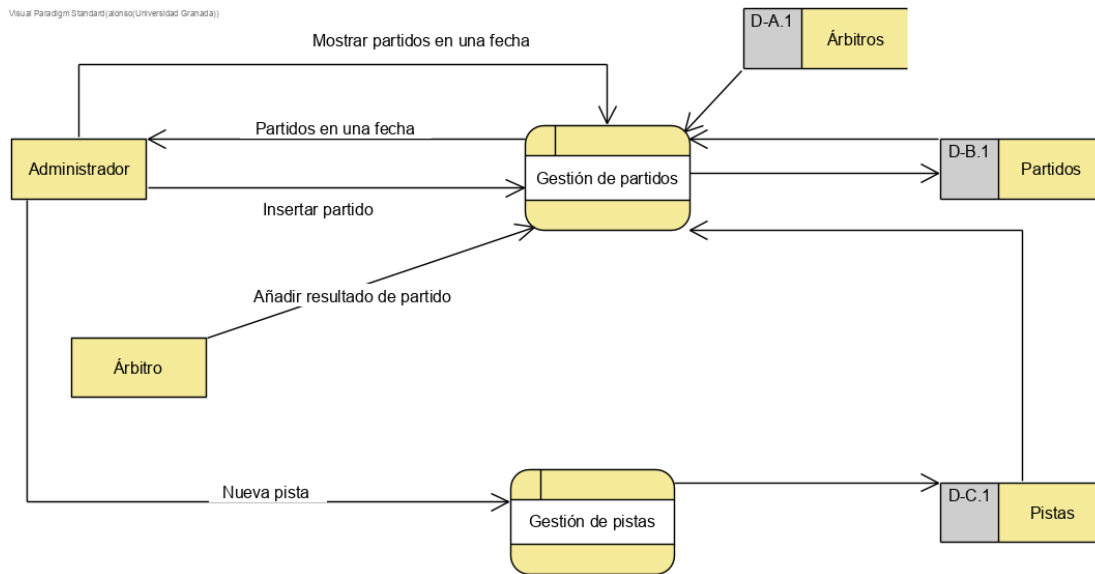
3.- Refinamientos:

3.1.- Refinamientos para el nivel 1:

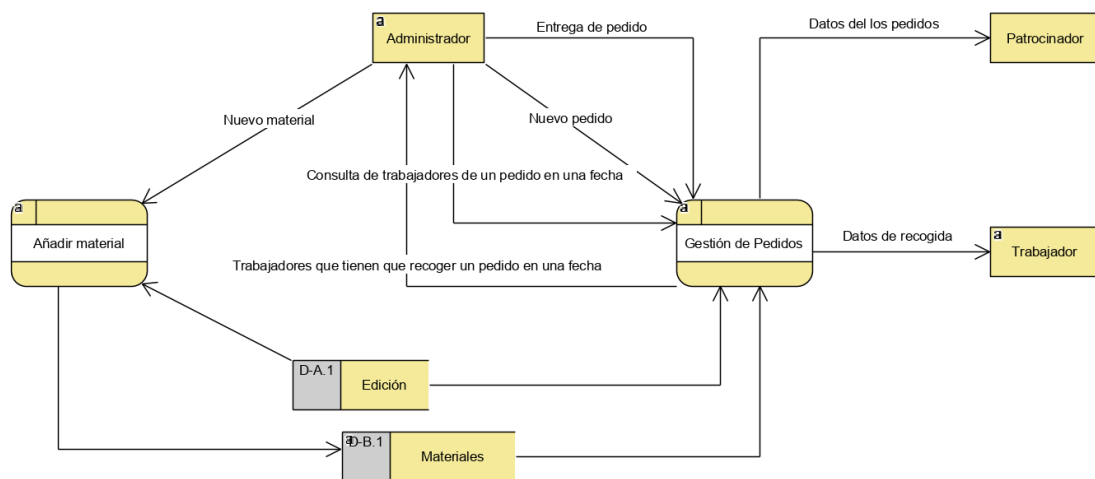
Proceso 1.-



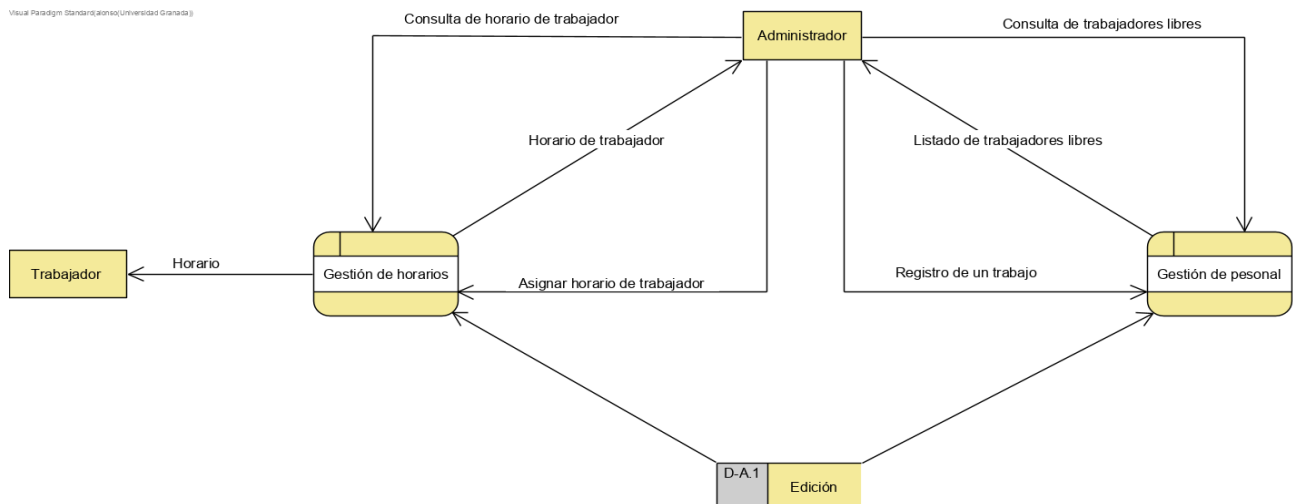
Proceso 2.-



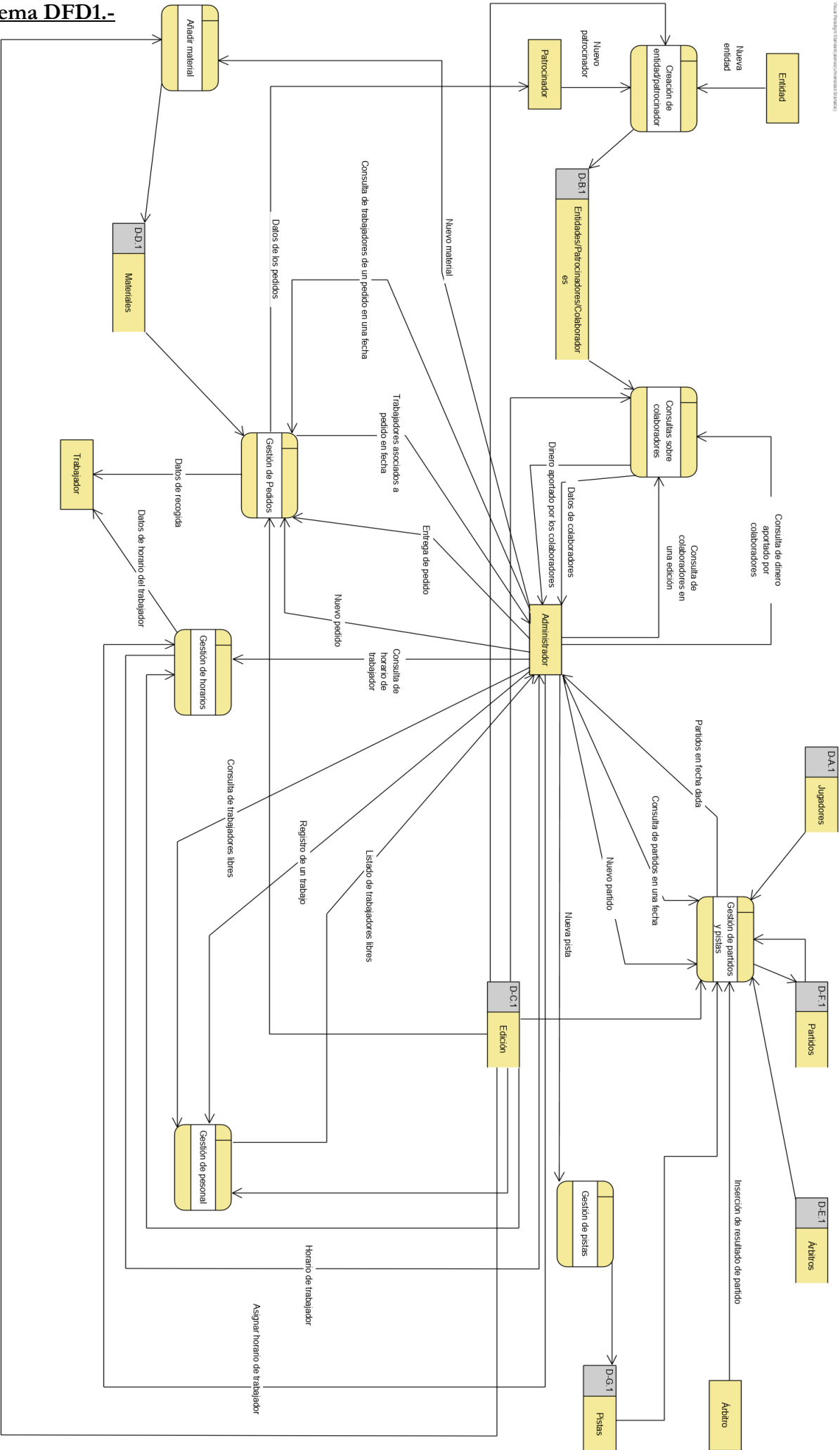
Proceso 3.-



Proceso 4.-

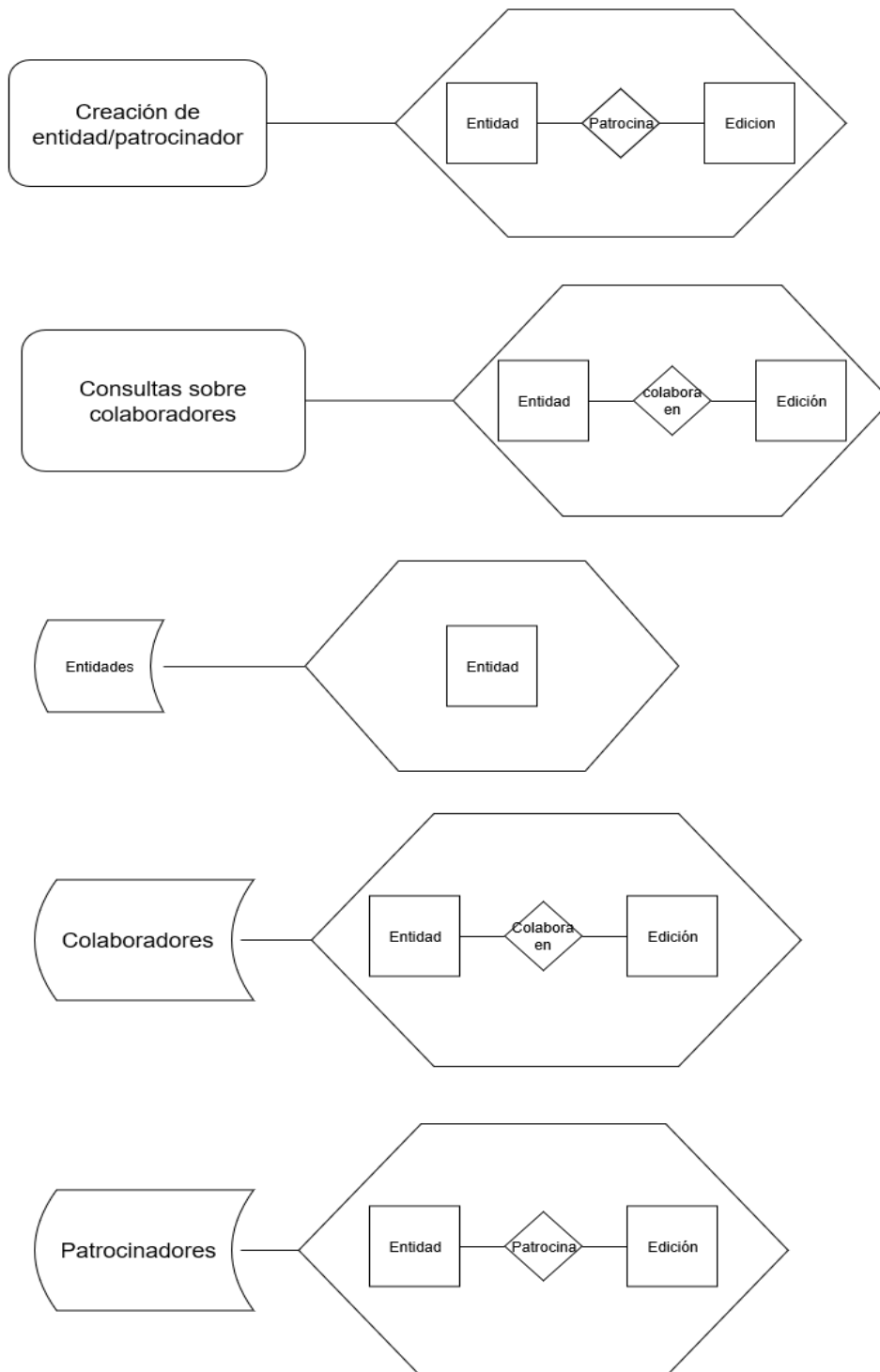


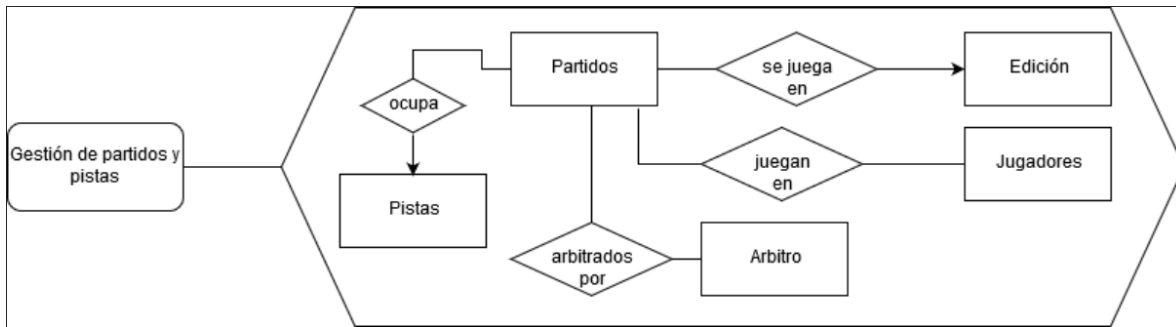
Esquema DFD1.-

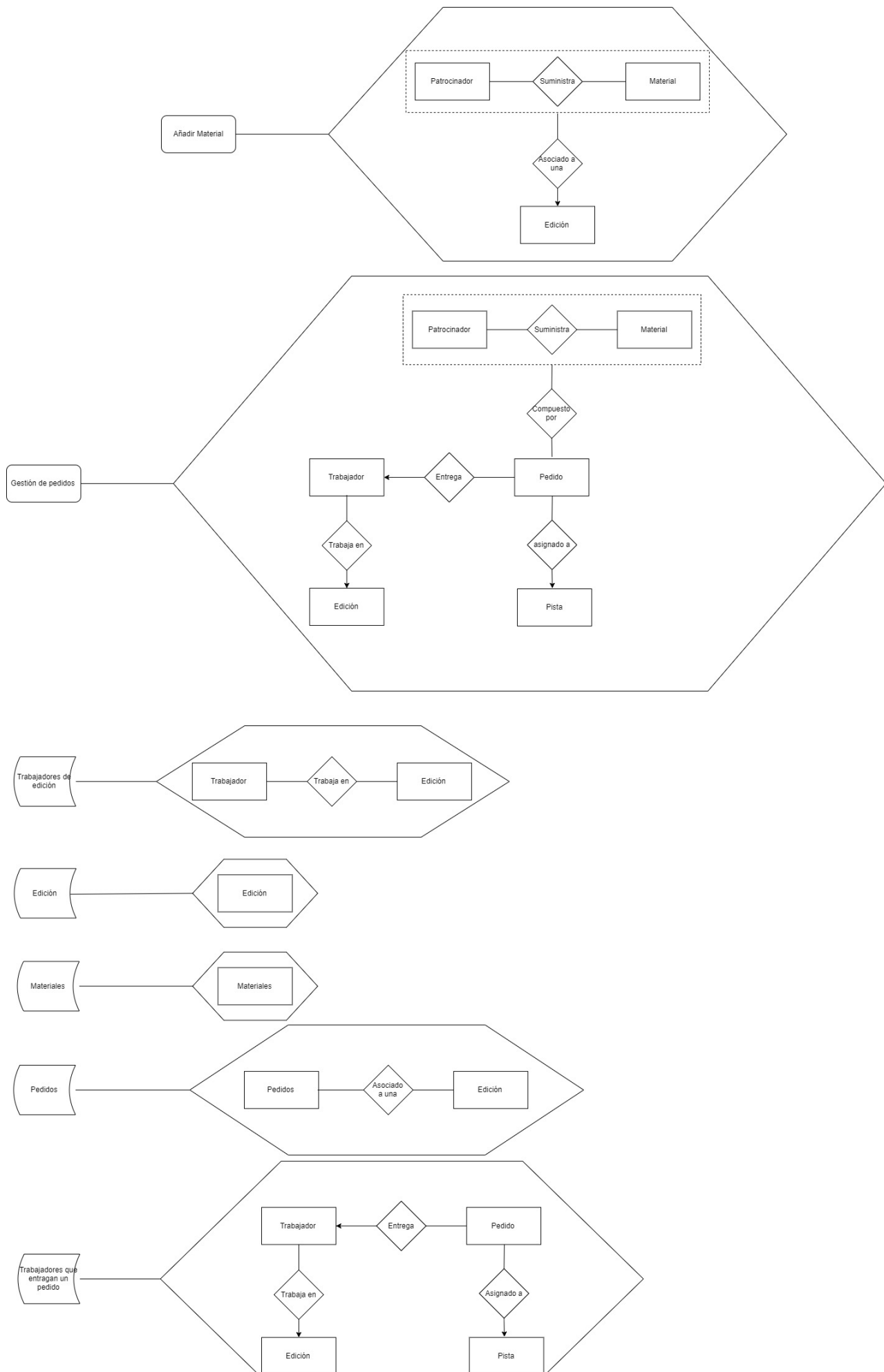


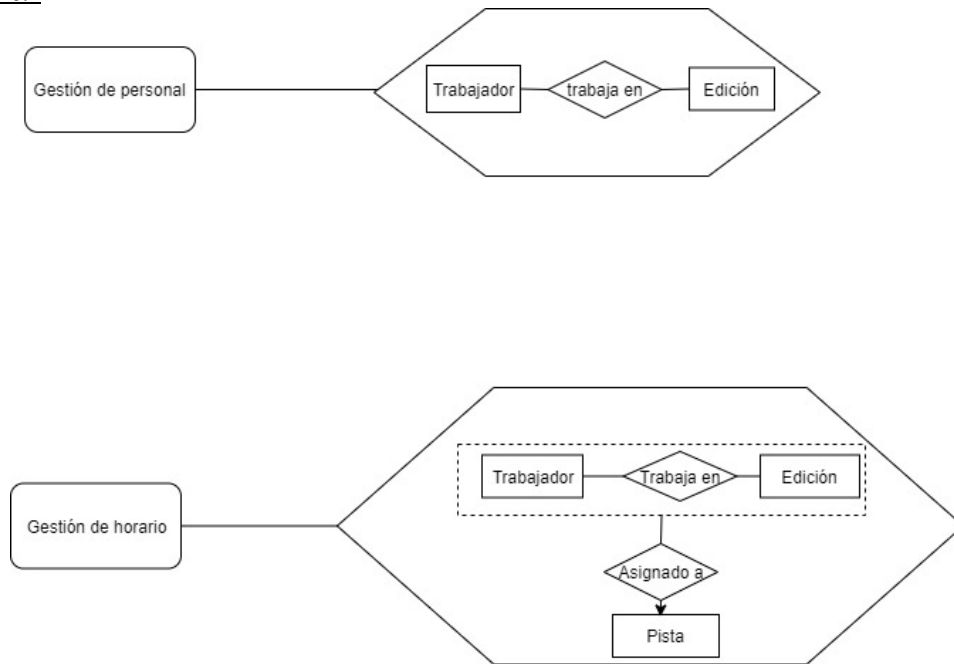
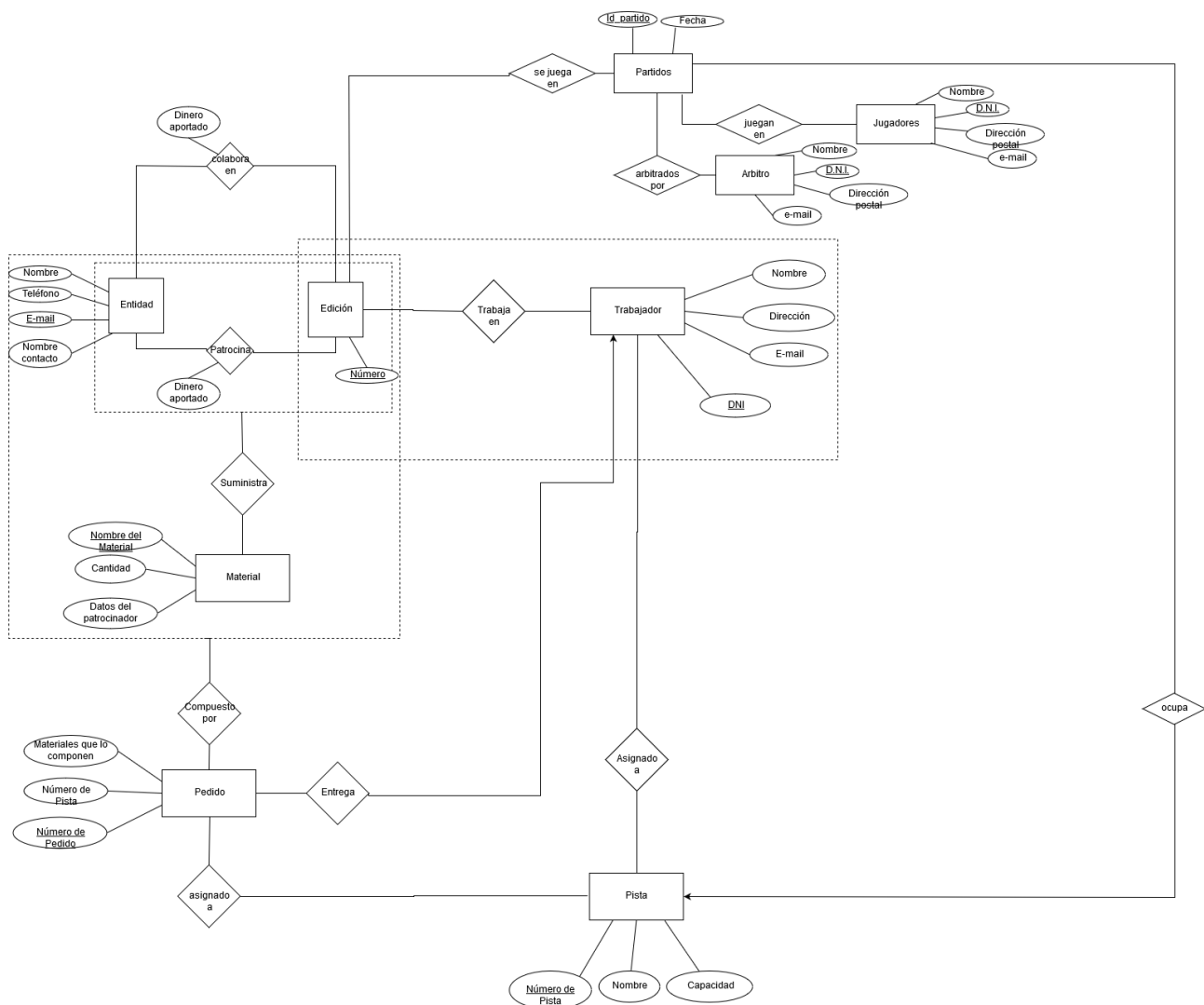
3.2.- Esquemas externos del nivel 1.

Subsistema 1.-



Subsistema 2.-**Subsistema 3.-***(página siguiente)*

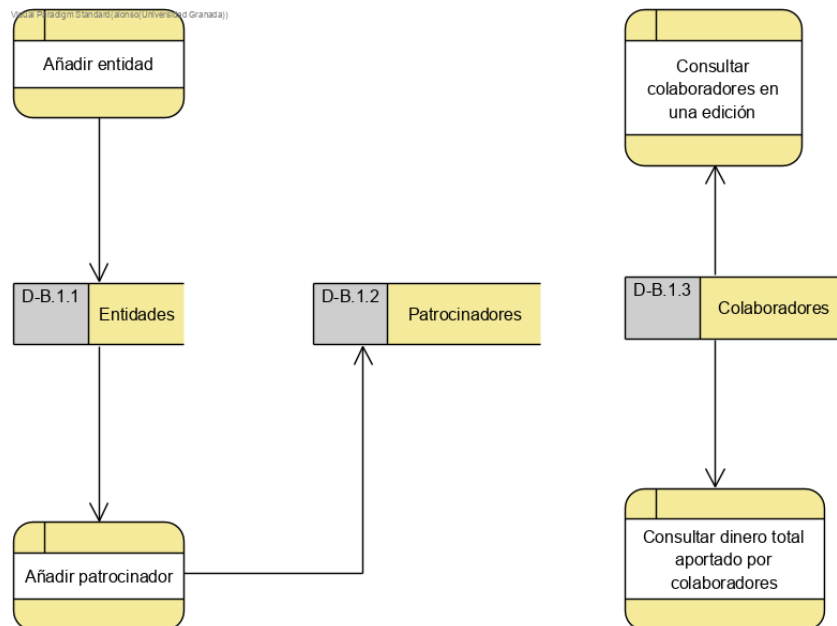


Subsistema 4.-**ER para el nivel 1.-**

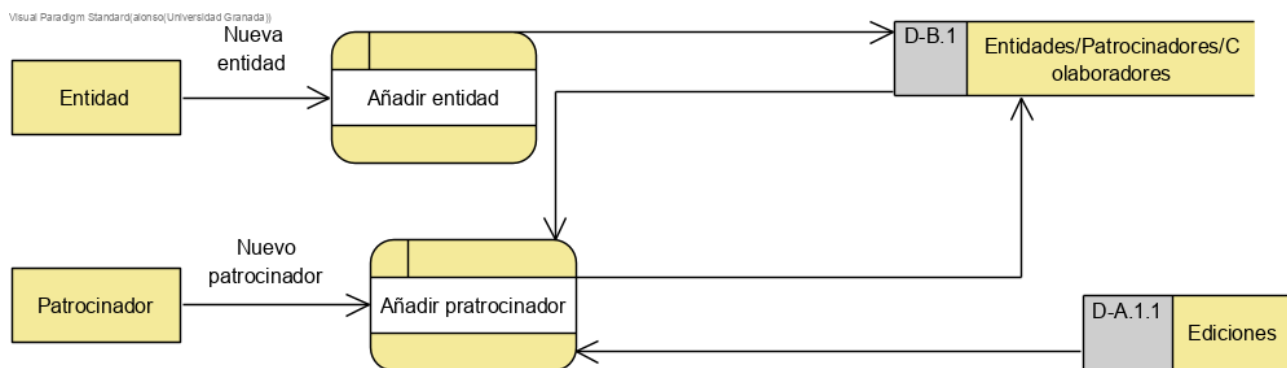
3.1.- Refinamientos para el nivel 2:

Subsistema 1.-

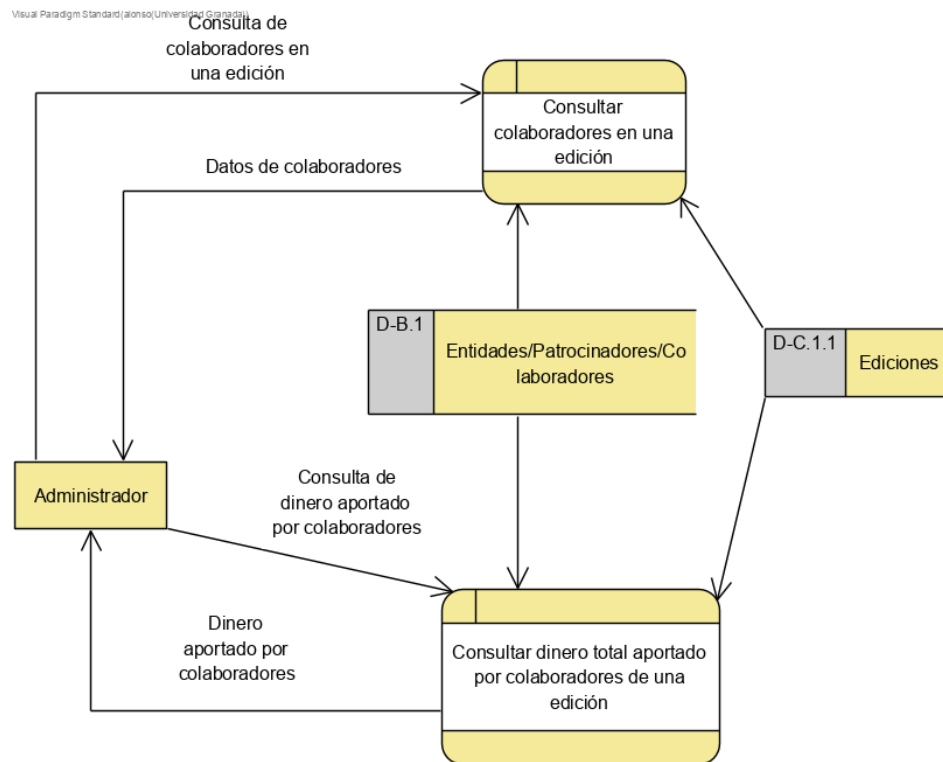
Refinamiento de almacén: Entidades/Patrocinadores/Colaboradores



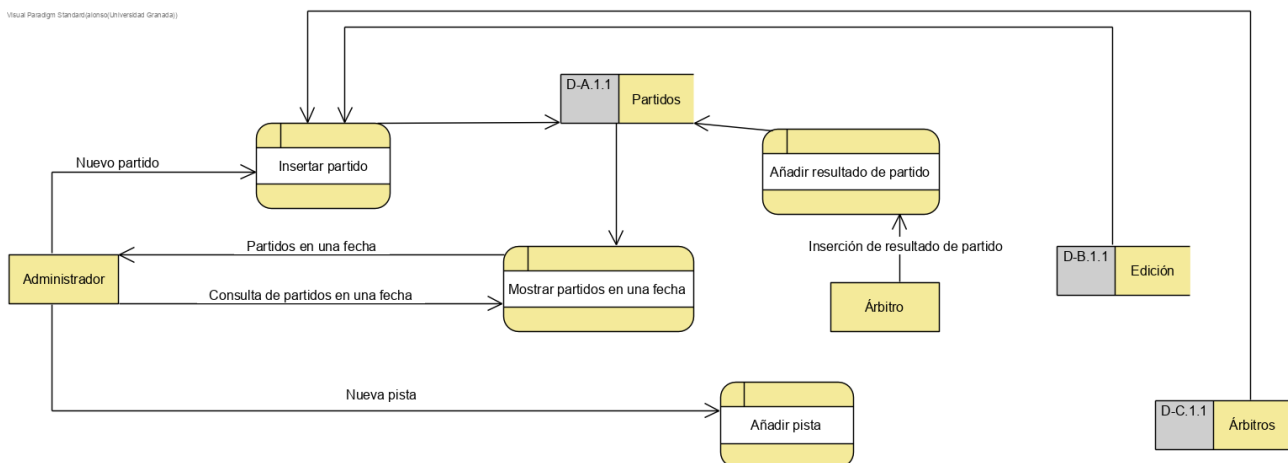
Refinamiento del proceso de creación de entidad/patrocinador



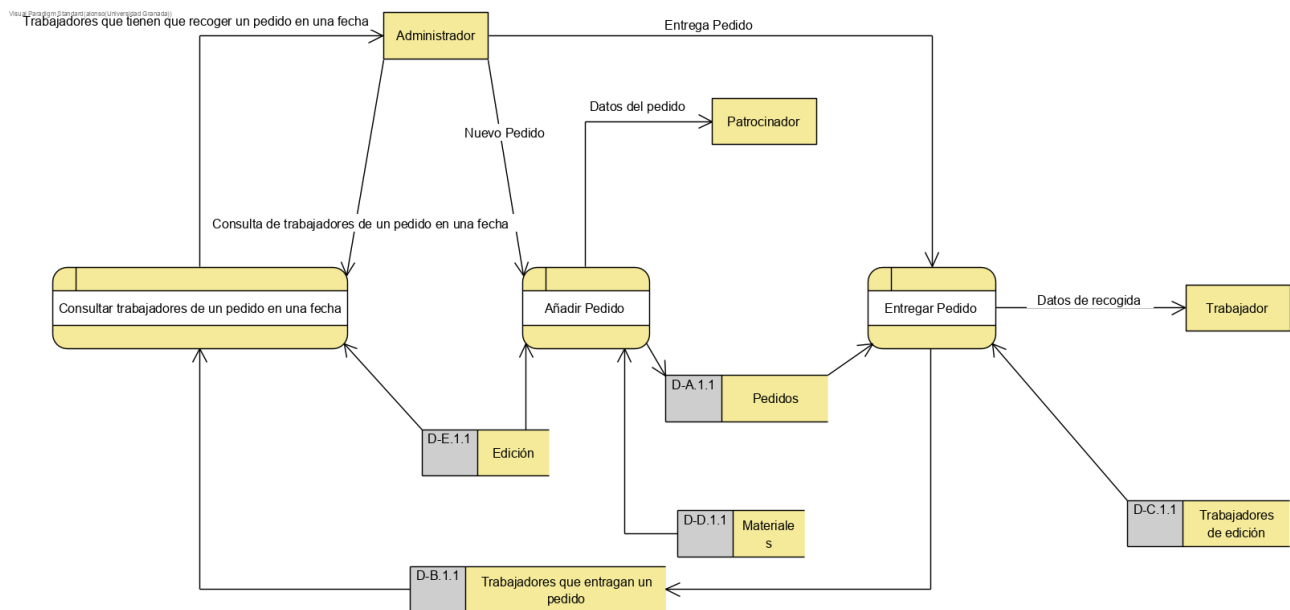
Refinamiento de consultas



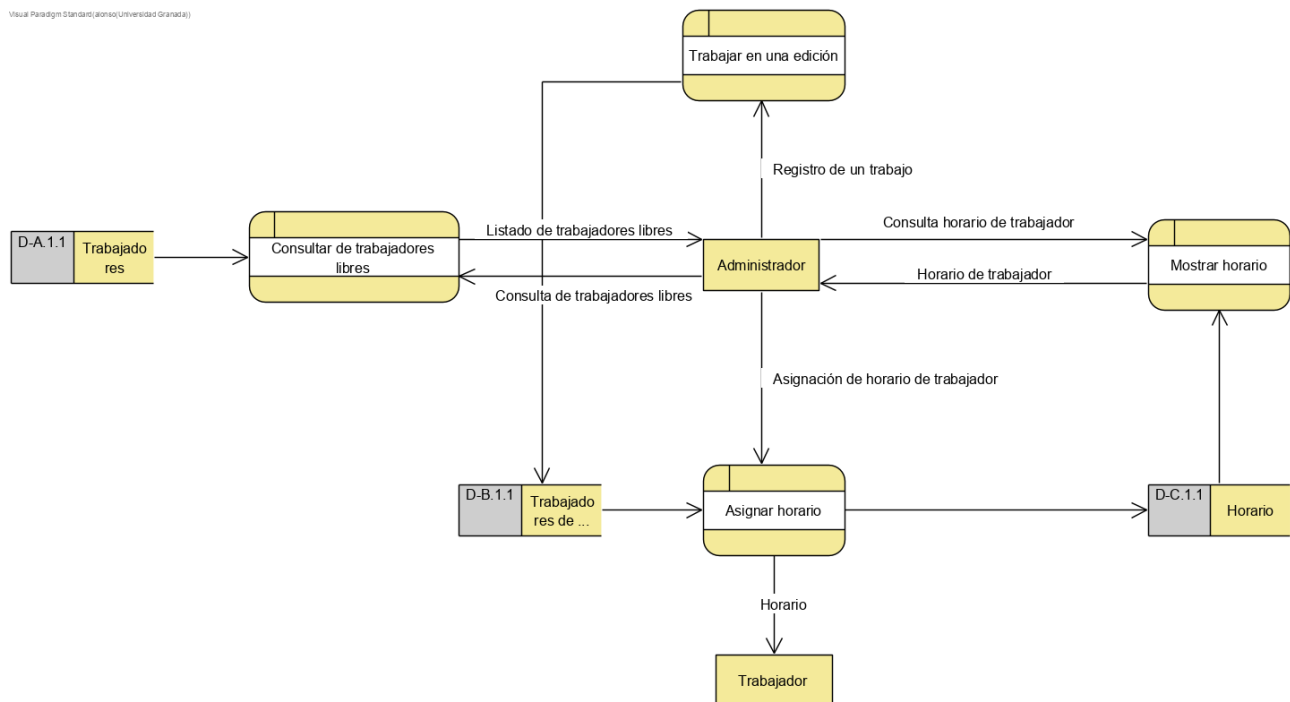
Subsistema 2.-



Subsistema 3.-

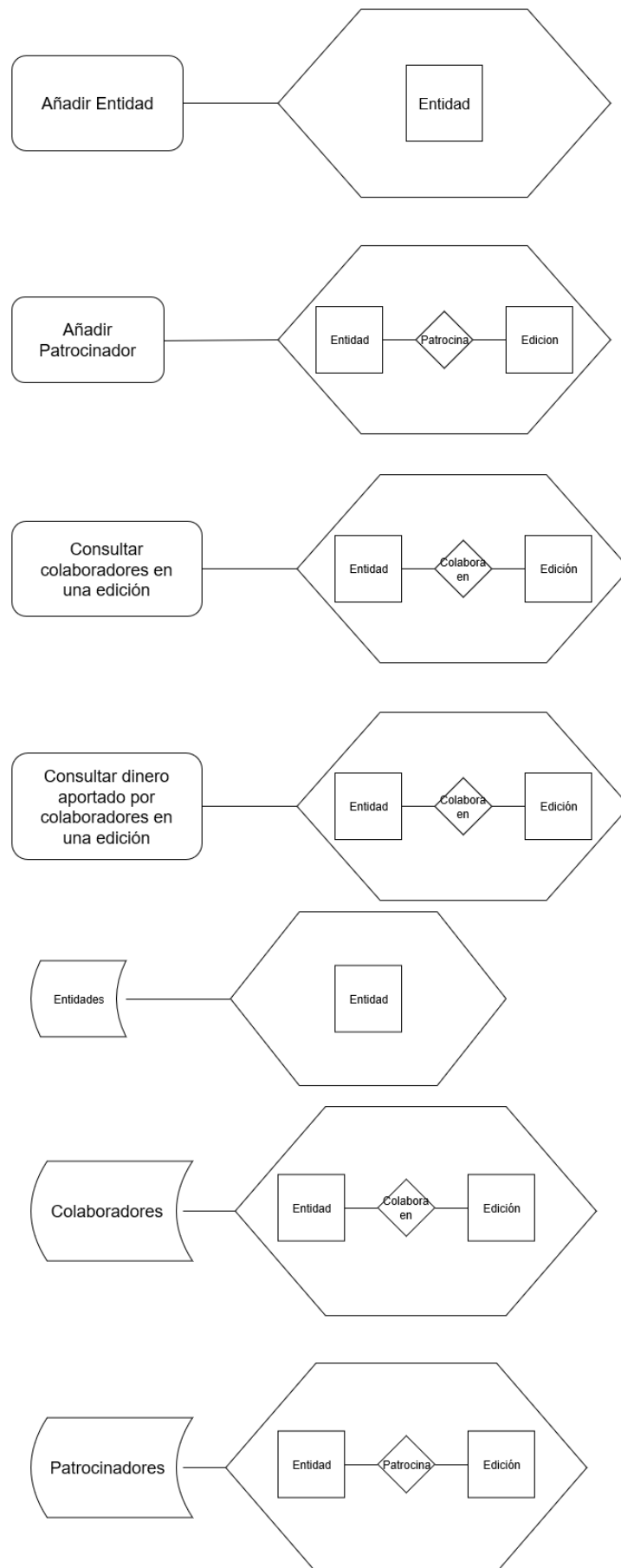


Subsistema 4.-

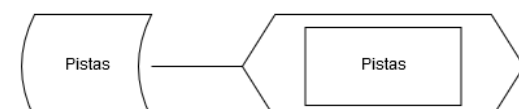
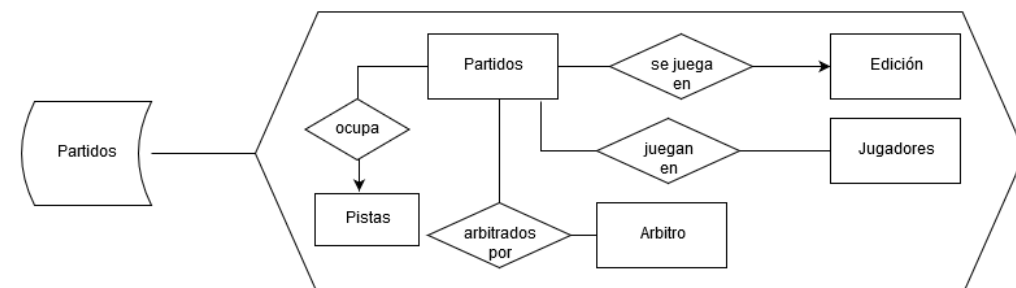
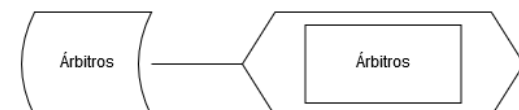
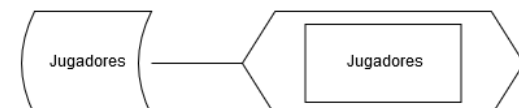
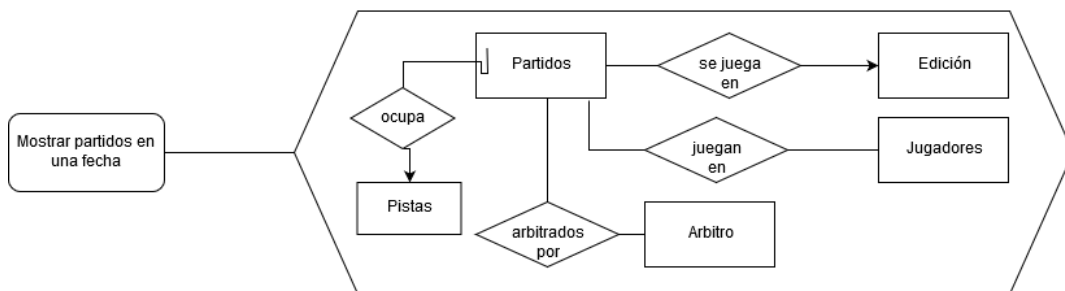
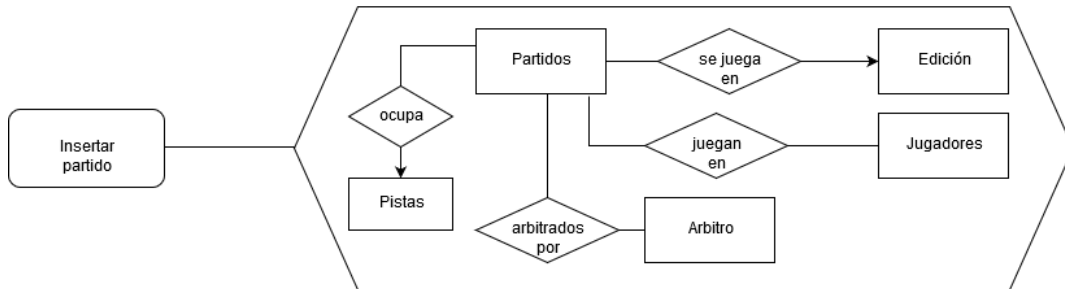
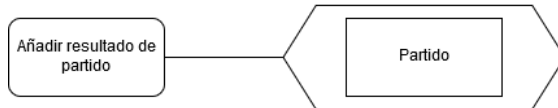
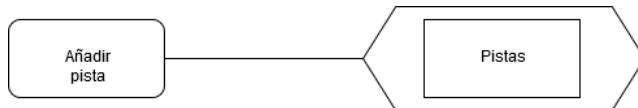


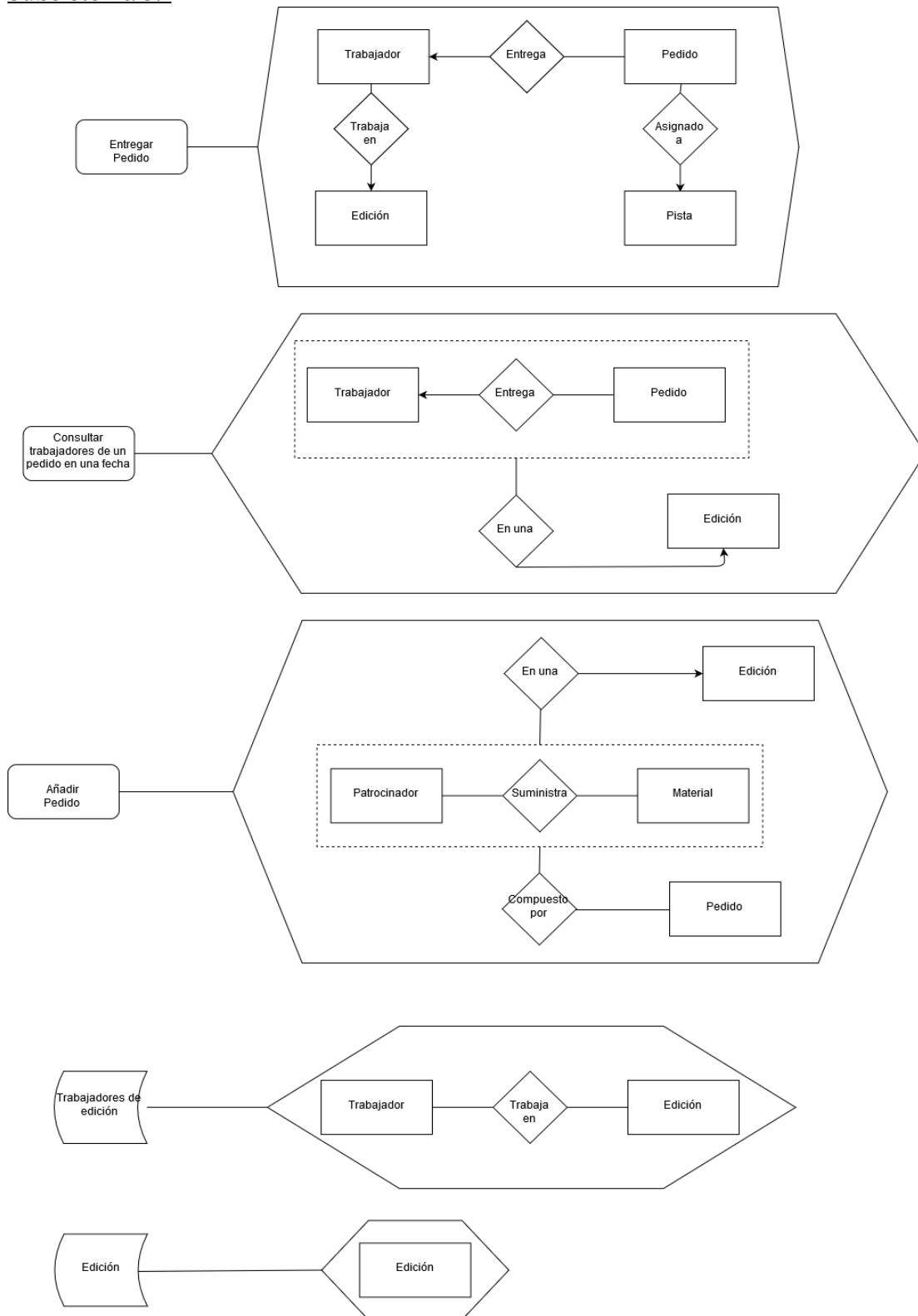
3.2.- Esquemas externos para el nivel 2:

Subsistema 1.-



Subsistema 2.-

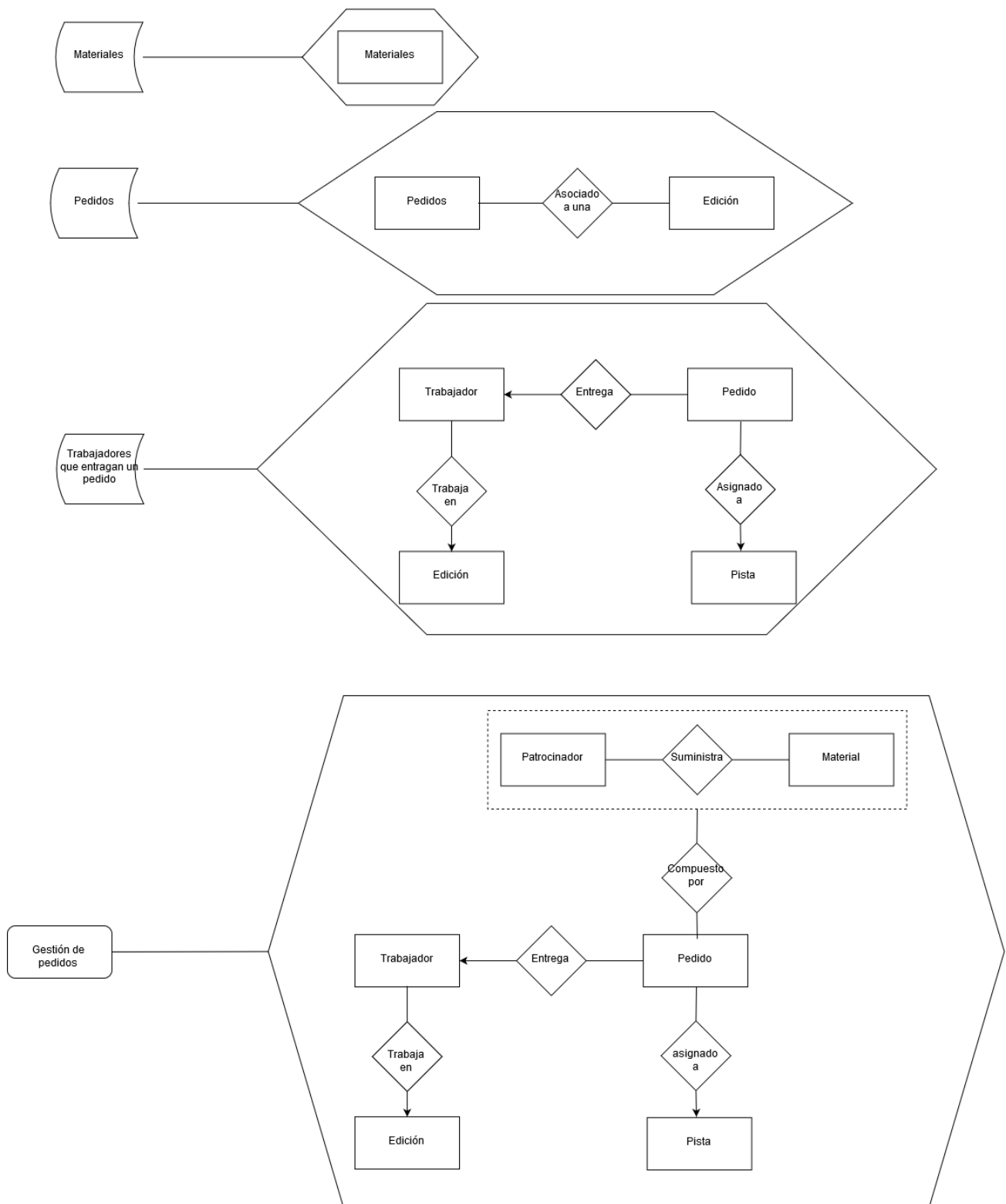


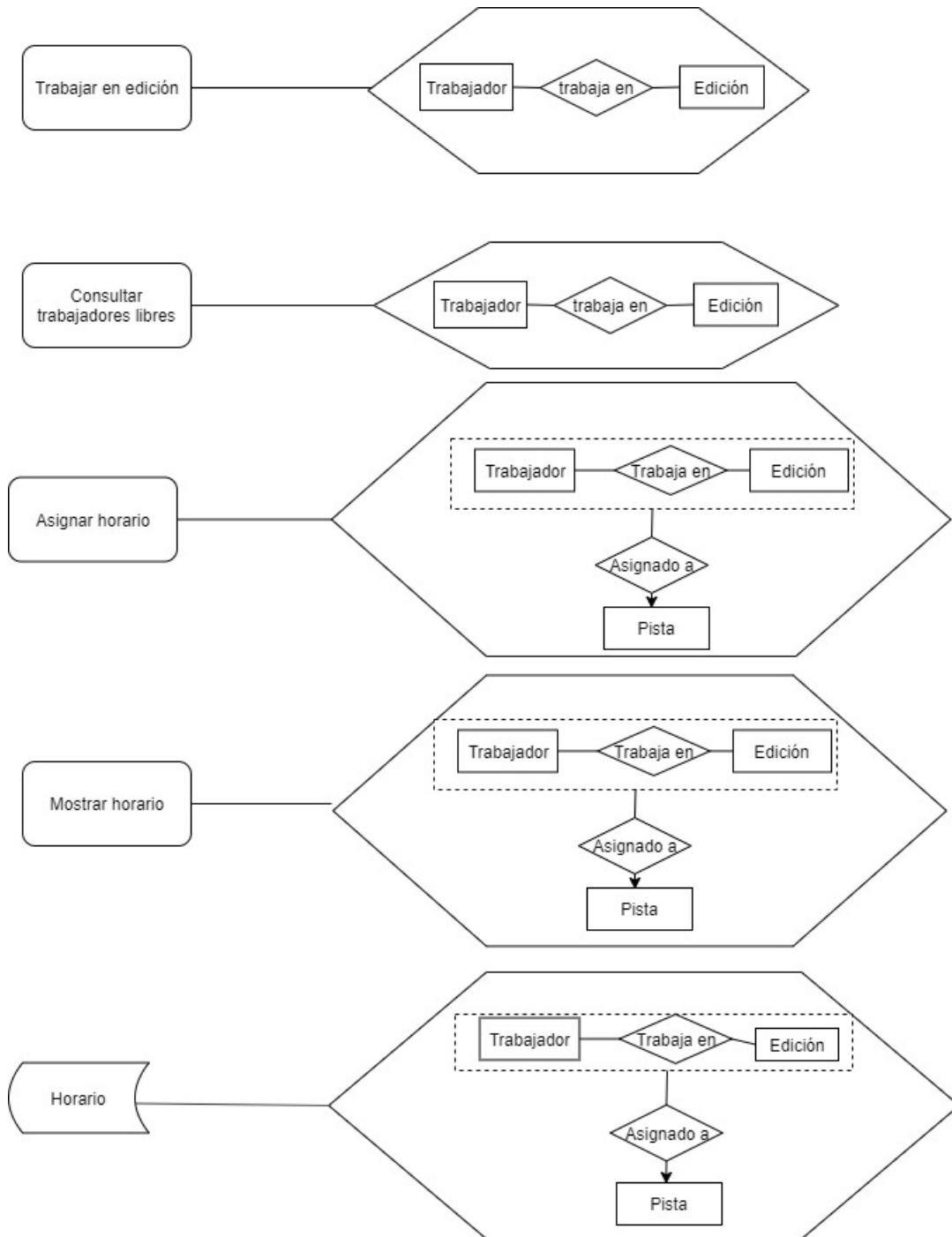
Subsistema 3.-

...

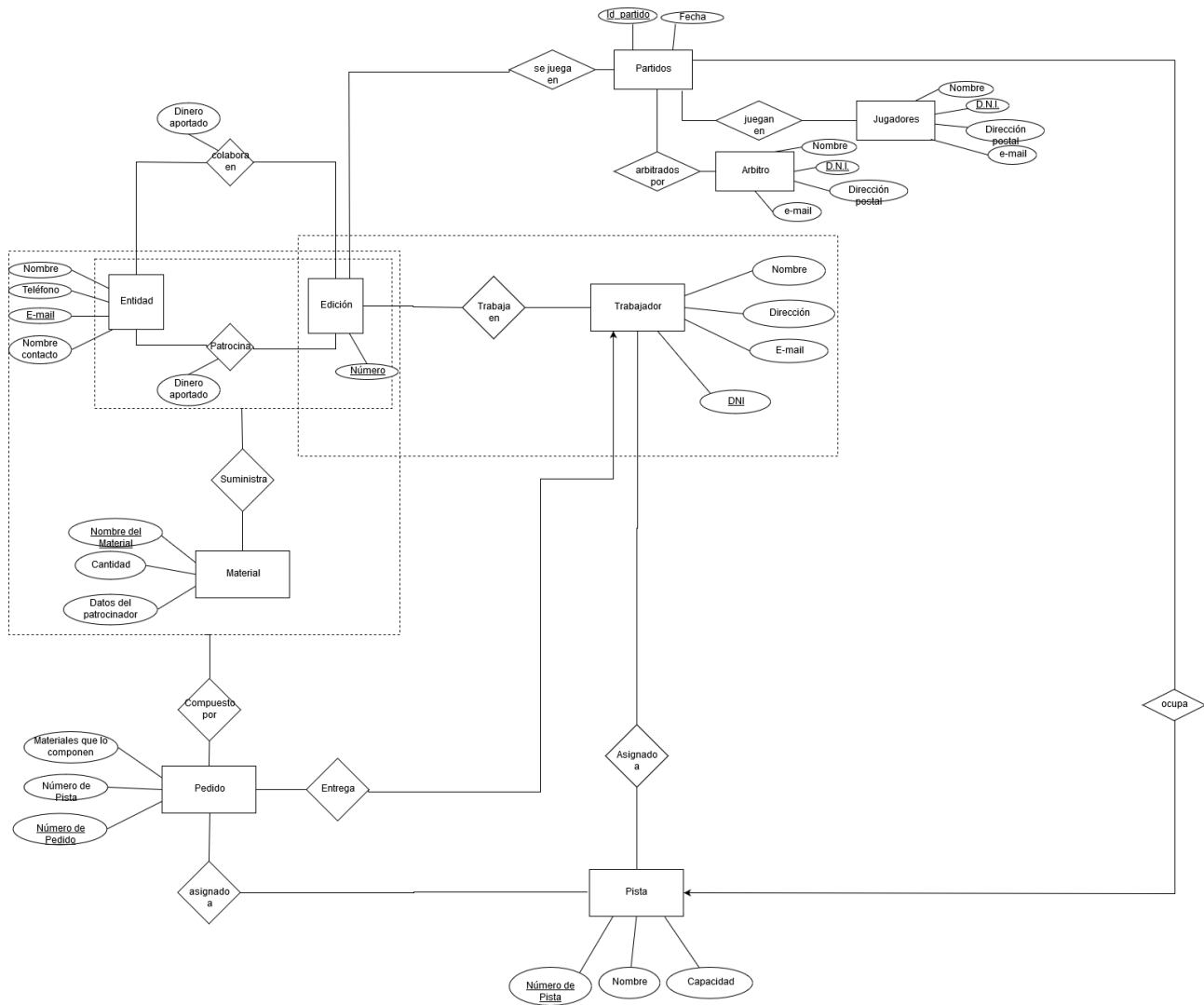
(este esquema externo continúa en la página siguiente)

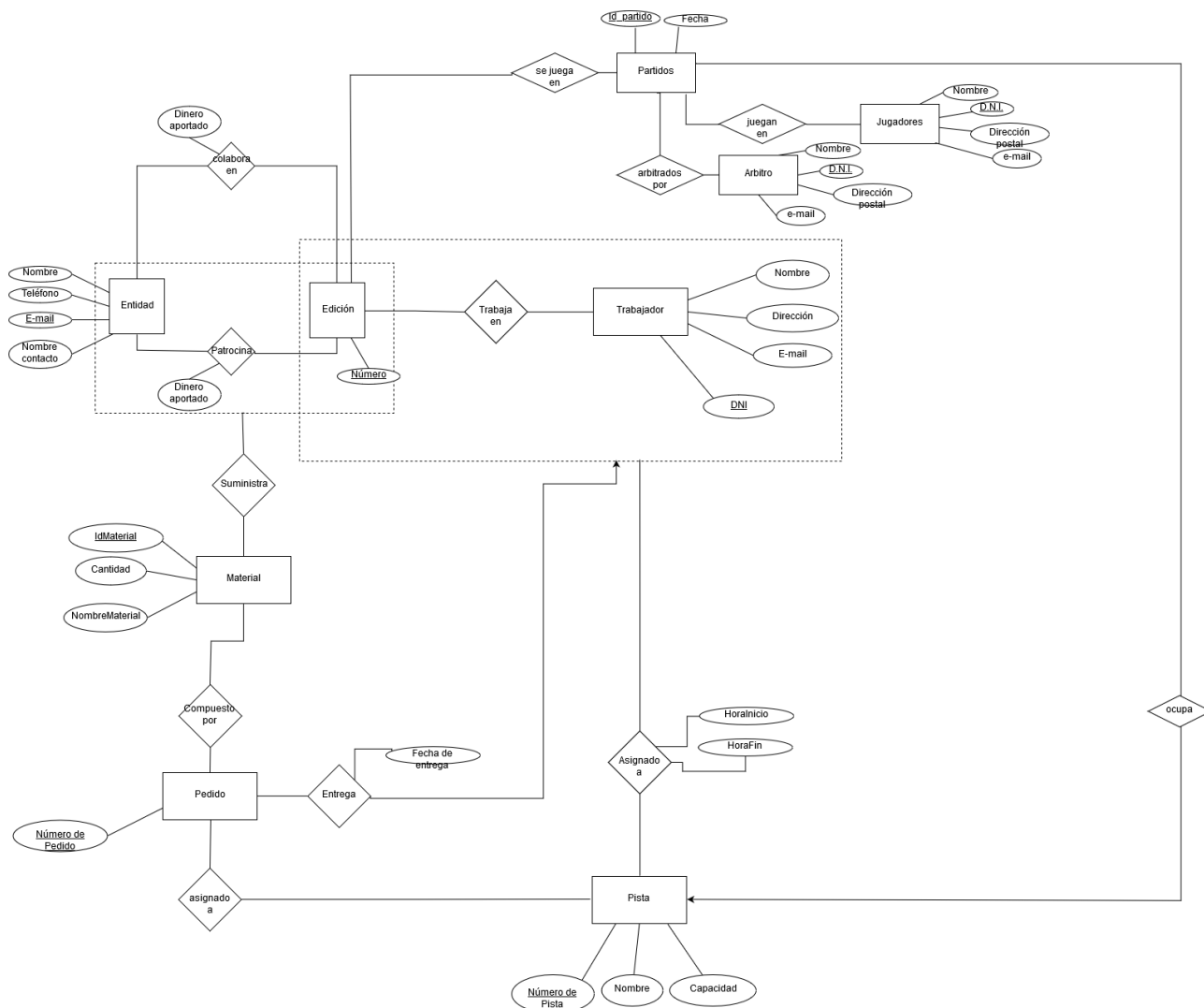
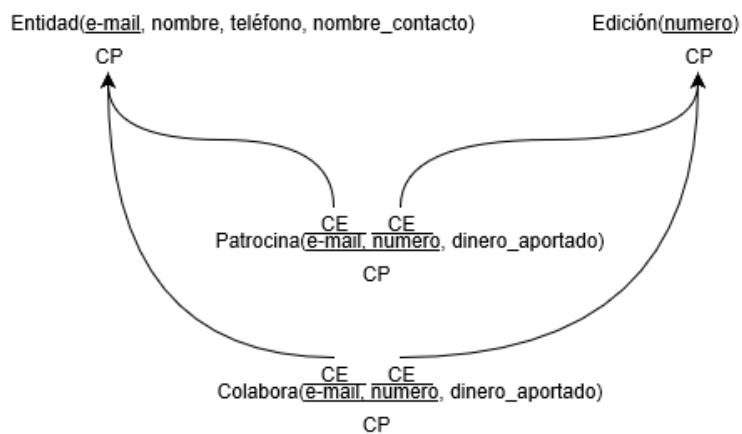
...



Subsistema 4.-

ER para el nivel 2.-



4.- Esquema Entidad-Relación con atributos:**5.2 Paso a tablas****SUBSISTEMA 1:**

SUBSISTEMA 2:

- (1) Partidos (id_partido, fecha)
CP
- (2) Jugadores (dni, e_mail, direccion_postal, nombre)
CP
- (3) Arbitro (dni, nombre, direccion_postal, e_mail)
CP
- (4) Se_juega_en (id_partido, numero_edicion)
CP CP
- (5) Arbitrados_por (dni, id_partido)
CP CP
- (6) Juegan_en (id_partido, dni)
CP CP
- (7) Ocupa (id_partido, id_pista)
CP
- (8) Edicion (numero)
CP
- (9) Pista (numeropista, nombre, capacidad)
CP

(resto de subsistemas en la página siguiente)

SUBSISTEMAS 3 y 4.-

(1) Material (IdMaterial, Cantidad, NombreMaterial)
CP

(2) Pedido (NumeroDePedido)
CP

(3) Trabajador (Nombre, Dirección, Email,
DNI)
CP

CE(1) CE(2)
(4) CompuestoPor (IdMaterial, NumeroDePedido)
CP

CE(TrEn) CE(TrEn) CE(1)
(5) Suministra(Email, Número, IdMaterial)
CP

CE(Edici) CE(3)
(6) TrabajaEn (Número, DNI)
CP

CE(2) CE(Pista)
(7) PedidoAsignadoA (NumeroDePedido, NumeroDePista)
CP

CE(2) CE(TrEn) CE(TrEn)
(8) Entrega (NumeroDePedido, FechaEntrega, Número, DNI)
CP

CE (edi) CE (6)
(9) TrabajadorAsignadoA (NumeroPista, Numero, Dni, Horainicio, HoraFin, Fecha)
CP

6 Implementación

6.1 Diseño físico de la base de datos: sentencias de creación de tablas

```
CREATE TABLE Edicion(  
Numero integer PRIMARY KEY);  
  
--Tablas subsistema 1:  
  
CREATE TABLE Entidad(  
email varchar2(50) CONSTRAINT Correo_no_tiene_Aroba CHECK (email like '%@%') PRIMARY KEY,  
nombre varchar2(40),  
telefono varchar2(12) CONSTRAINT Telefono_no_valido CHECK (regexp_like(telefono, '[  
+"[0-9]*[0-9]*"?[0-9]{9}')),  
contacto varchar2(100));  
  
CREATE TABLE Patrocina(  
email REFERENCES Entidad(email) NOT NULL,  
numero_edicion references Edicion(numero) NOT NULL,  
dinero_aporteado float,  
PRIMARY KEY(email, numero_edicion)  
);  
  
CREATE TABLE Colabora(  
email REFERENCES Entidad(email) NOT NULL,  
numero_edicion REFERENCES Edicion(numero) NOT NULL,  
dinero_aporteado float,  
PRIMARY KEY (email, numero_edicion)  
);  
  
--Tablas subsistema 2:  
  
create table partidos (  
    id_partido integer primary key,  
    fecha date  
);  
  
create table jugadores (  
    e_mail varchar2(50) not null,  
    direccion_postal varchar2(70),  
    nombre varchar2(40) ,  
    dni varchar2(10) primary key  
);  
  
create table arbitro (  
    dni varchar2(10) primary key,  
    nombre varchar2(40),  
    direccion_postal varchar2(70),
```



```

    e_mail varchar2(50) not null constraint email_ok check (e_mail like '%@%') );

create table se_juega_en (
    id_partido references partidos(id_partido) ,
    numero_edicion references Edicion(Numero) ,
    primary key (id_partido, numero_edicion)
);

create table arbitrados_por (
    dni references arbitro(dni) primary key,
    id_partido references partidos(id_partido) );

create table juegan_en (
    id_partido references partidos(id_partido) ,
    dni references jugadores(dni) ,
    primary key (id_partido, dni)
);

CREATE TABLE Pista(
    numeropista NUMBER(4) CONSTRAINT valores_mayor_a_0 CHECK (numeropista > 0 ) PRIMARY KEY,
    nombre VARCHAR2(40) NOT NULL,
    capacidad NUMBER(4) NOT NULL
);

create table ocupa (
    id_partido references partidos (id_partido) primary key,
    id_pista REFERENCES pista(numeropista)
);

-- Tablas subsistema 3 (y parte del 4):

CREATE TABLE Material(
    idmaterial VARCHAR2(10) PRIMARY KEY,
    cantidad NUMBER(4) NOT NULL,
    nombrematerial VARCHAR2(40) NOT NULL
);

CREATE TABLE Pedido(
    numerodepedido NUMBER(4) CONSTRAINT valores_mayor_que_0 CHECK (numerodepedido > 0 ) PRIMARY KEY
);

CREATE TABLE Trabajador(
    dni VARCHAR2(9) CONSTRAINT DNI_no_valido CHECK (REGEXP_LIKE (dni, '[0-9]{8}[A-Z]')) PRIMARY KEY,
    nombre VARCHAR2(40) NOT NULL,
    telefono NUMBER(9),
    email VARCHAR2(50) CONSTRAINT email_no_valido CHECK (email like '%@%')
);

```

```
CREATE TABLE Suministra(  
    email not null,  
    numeros not null,  
    idmaterial REFERENCES Material(idmaterial),  
    FOREIGN KEY (email, numeros) REFERENCES Patrocina (email, numero_edicion),  
    PRIMARY KEY (email, numeros, idmaterial)  
);  
  
CREATE TABLE CompuestoPor(  
    idmaterial REFERENCES Material(idmaterial),  
    numerodepedido REFERENCES Pedido(numerodepedido),  
    PRIMARY KEY (idmaterial, numerodepedido)  
);  
  
CREATE TABLE TrabajaEn(  
    numero REFERENCES Edicion (numero),  
    dni REFERENCES Trabajador(dni),  
    PRIMARY KEY (numero, dni)  
);  
  
CREATE TABLE Entrega(  
    numerodepedido REFERENCES Pedido(numerodepedido),  
    fechaentrega DATE,  
    numero NOT NULL,  
    dni NOT NULL,  
    FOREIGN KEY (numero, dni) REFERENCES TrabajaEn (numero, dni),  
    PRIMARY KEY (numerodepedido, fechaentrega)  
);  
  
CREATE TABLE PedidoAsignadoA(  
    numerodepedido REFERENCES Pedido (numerodepedido),  
    numeropista REFERENCES Pista (numeropista),  
    PRIMARY KEY (numerodepedido, numeropista)  
);  
  
-- Tablas subsistema 4:  
  
CREATE TABLE TrabajadorAsignadoA(  
    numeropista REFERENCES Pista (numeropista),  
    numero NOT NULL,  
    dni NOT NULL,  
    horainicio DATE NOT NULL,  
    horafin DATE NOT NULL,  
    fecha DATE NOT NULL,  
    FOREIGN KEY (numero, dni) REFERENCES TrabajaEn (numero, dni),  
    PRIMARY KEY (numeropista, numero, dni)  
);
```

6.2 Inserción de datos de prueba

```
-- inserts subsistema 1:

INSERT INTO Edicion (numero) VALUES (1);
INSERT INTO Edicion (numero) VALUES (2);
INSERT INTO Edicion (numero) VALUES (3);
INSERT INTO Edicion (numero) VALUES (4);
INSERT INTO Edicion (numero) VALUES (5);
INSERT INTO Edicion (numero) VALUES (6);
INSERT INTO Edicion (numero) VALUES (7);
INSERT INTO Edicion (numero) VALUES (8);
INSERT INTO Edicion (numero) VALUES (9);
INSERT INTO Edicion (numero) VALUES (10);

INSERT INTO Entidad (email, nombre, telefono, contacto) VALUES ('constructoresmanolo@gmail.com', 'Constructora Manolo', '32619786543', 'Manolo');
INSERT INTO Entidad (email, nombre, telefono, contacto) VALUES ('peluqueriaantonio@gmail.com', 'Peluqueria Antonio', '958345678', 'Antonio');
INSERT INTO Entidad (email, nombre, telefono, contacto) VALUES ('comercialesmaria_7877@gmail.com', 'Veinte Duros', '+32722435678', 'Josefa');
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('comercialesmaria_7877@gmail.com', 1, 545.67);
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('peluqueriaantonio@gmail.com', 1, 545.67);
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('comercialesmaria_7877@gmail.com', 2, 180.56);
INSERT INTO Colabora (email, numero_edicion, dinero_aporteado) VALUES ('constructoresmanolo@gmail.com', 1, 90.86);
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('comercialesmaria_7877@gmail.com', 3, 234.78);
--Hata aqui deberian de funcionar todos los INSERT

--Estas inserciones son las que quiero probar:
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('constructoresmanolo@gmail.com', 4, 326.58);
INSERT INTO Colabora (email, numero_edicion, dinero_aporteado) VALUES ('comercialesmaria_7877@gmail.com', 4, 326.58);
INSERT INTO Colabora (email, numero_edicion, dinero_aporteado) VALUES ('peluqueriaantonio@gmail.com', 4, 487.20);

--Las inserciones justo debajo deberían de hacer saltar los disparadores:
/*
INSERT INTO Colabora (email, numero_edicion, dinero_aporteado) VALUES ('constructoresmanolo@gmail.com', 4, 326.58);
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('comercialesmaria_7877@gmail.com', 4, 326.58);
INSERT INTO Patrocina (email, numero_edicion, dinero_aporteado) VALUES ('peluqueriaantonio@gmail.com', 4, 487.20);
*/
```

```

-- INSERT subsistema 2:

INSERT INTO Pista (numeropista, nombre, capacidad) VALUES (1, 'PistaFelicianoLopez', 500);
INSERT INTO Pista (numeropista, nombre, capacidad) VALUES (2, 'PistaGrande', 800);
INSERT INTO Pista (numeropista, nombre, capacidad) VALUES (3, 'Pista3', 250);

insert into partidos (id_partido, fecha) VALUES (1, TO_DATE('12-01-2012','dd-mm-yyyy'));
insert into partidos (id_partido, fecha) VALUES (2, TO_DATE('03-09-2018','dd-mm-yyyy'));
insert into partidos (id_partido, fecha) VALUES (3, TO_DATE('23-12-2020','dd-mm-yyyy'));
insert into partidos (id_partido, fecha) VALUES (4, TO_DATE('23-12-2020','dd-mm-yyyy'));
insert into partidos (id_partido, fecha) VALUES (5, TO_DATE('23-12-2020','dd-mm-yyyy'));
insert into partidos (id_partido, fecha) VALUES (6, TO_DATE('23-12-2020','dd-mm-yyyy'));

insert into jugadores (nombre, dni, direccion_postal, e_mail) values ('Federico Martin Bahamontes', '12233412A', 'Plaza de Don Pardo 12', 'federbm@gmail.com');
insert into jugadores (nombre, dni, direccion_postal, e_mail) values ('Antonio Martin Garcia', '12998844A', 'Avenida de las Acacias 12 1Izda', 'antoniomg@gmail.com');
insert into jugadores (nombre, dni, direccion_postal, e_mail) values ('Luis Garcia Garcia', '24231212A', 'Calle General Redondo 12', 'luisitogarci@gmail.com');

insert into arbitro (dni, nombre, direccion_postal, e_mail) values ('33443322E', 'Maria Jose Merino Gutierrez', 'Paseo de San Rafael 5 2 B', 'marimegu@yahoo.es');
insert into arbitro (dni, nombre, direccion_postal, e_mail) values ('33449922E', 'Antonio Garcia Martin', 'Calle de Mendez Alvaro 5', 'antoningm@yahoo.es');
insert into arbitro (dni, nombre, direccion_postal, e_mail) values ('12123498J', 'Luis Gutierrez Aragon', 'Carretera de Extremadura 12 1 D', 'luisga@yahoo.es');

insert into se_juega_en (id_partido, numero_edicion) values (1, 1);

insert into arbitrados_por (dni, id_partido) values ('33443322E', 1);
insert into arbitrados_por (dni, id_partido) values ('33449922E', 2);

insert into juegan_en (id_partido, dni) values (2, '12233412A');

insert into ocupa (id_partido, id_pista) values (3,1) ;
insert into ocupa (id_partido, id_pista) values (4,1) ;
insert into ocupa (id_partido, id_pista) values (5,1) ;
insert into ocupa (id_partido, id_pista) values (6,1) ;

```

```
-- INSERT subsistema 3 y 4:

INSERT INTO Material (idmaterial, cantidad, nombrematerial) VALUES ('124387cds', 12, 'pelotas');
INSERT INTO Material (idmaterial, cantidad, nombrematerial) VALUES ('65879dfg', 7, 'redes');
INSERT INTO Material (idmaterial, cantidad, nombrematerial) VALUES ('239dfrgg', 4, 'raquetas');

INSERT INTO Pedido (numerodepedido) VALUES (1);
INSERT INTO Pedido (numerodepedido) VALUES (2);
INSERT INTO Pedido (numerodepedido) VALUES (3);
INSERT INTO Pedido (numerodepedido) VALUES (4);

INSERT INTO Trabajador (dni, nombre, telefono, email) VALUES ('12345678K', 'Pepito', 611345672, 'pepito@cgmail.com');
INSERT INTO Trabajador (dni, nombre, telefono, email) VALUES ('11342678F', 'Antonio', 621326572, 'antonio23@cgmail.com');
INSERT INTO Trabajador (dni, nombre, telefono, email) VALUES ('11223344F', 'Pepe', 632126212, 'pepe123@cgmail.com');

-- LOS INSERT asociados a PISTA están en la zona del S2 (necesarios)

INSERT INTO Suministra (email, numeros, idmaterial) VALUES ('comercialesmaria_7877@gmail.com', 2, '124387cds');
INSERT INTO Suministra (email, numeros, idmaterial) VALUES ('peluqueriaantonio@gmail.com', 1, '65879dfg');
INSERT INTO Suministra (email, numeros, idmaterial) VALUES ('peluqueriaantonio@gmail.com', 1, '239dfrgg');

INSERT INTO CompuestoPor (idmaterial, numerodepedido) VALUES ('124387cds', 1);
INSERT INTO CompuestoPor (idmaterial, numerodepedido) VALUES ('65879dfg', 2);

INSERT INTO TrabajaEn (numero, dni) VALUES (3, '12345678K');
INSERT INTO TrabajaEn (numero, dni) VALUES (2, '11342678F');
INSERT INTO TrabajaEn (numero, dni) VALUES (2, '11223344F');

INSERT INTO Entrega (numerodepedido, fechaentrega, numero, dni) VALUES (1, TO_DATE('12/03/2019','dd/mm/yyyy'), 3, '12345678K');
INSERT INTO Entrega (numerodepedido, fechaentrega, numero, dni) VALUES (2, TO_DATE('25/05/2020','dd/mm/yyyy'), 2, '11342678F');

INSERT INTO PedidoAsignadoA (numerodepedido, numeropista) VALUES (1, 1);
INSERT INTO PedidoAsignadoA (numerodepedido, numeropista) VALUES (4, 3);

INSERT INTO TrabajadorAsignadoA (numeropista, numero, dni, horainicio, horafin, fecha) VALUES (1, 3, '12345678K', TO_DATE('12/03/2019 11:30:00','dd/mm/yyyy hh:mi:ss'), TO_DATE('12/03/2019 12:00:00','dd/mm/yyyy hh:mi:ss'), TO_DATE('12/03/2019','dd/mm/yyyy'));
```

```

INSERT INTO TrabajadorAsignadoA (numeropista, numero, dni, horainicio, horafin, fecha) VALUES (2, 2, '11342678F', TO_DATE('10/01/2020 07:30:00','dd/mm/yyyy hh:mi:ss'), TO_DATE('10/01/2020 08:45:00','dd/mm/yyyy hh:mi:ss'), TO_DATE('10/01/2020','dd/mm/yyyy'));

--Insert que hace saltar el disparador 3
INSERT INTO CompuestoPor (idmaterial, numerodepedido) VALUES ('239dfrrg', 1);

--Insert que hace saltar el disparador 4

INSERT INTO TrabajadorAsignadoA (numeropista, numero, dni, horainicio, horafin, fecha) VALUES (2, 2, '11342678F', TO_DATE('10/01/2020 11:30:00','dd/mm/yyyy hh:mi:ss'), TO_DATE('10/01/2020 12:45:00','dd/mm/yyyy hh:mi:ss'), TO_DATE('10/01/2020','dd/mm/yyyy'));

```

6.3 Disparadores

6.3.1 Subsistema 1

```

CREATE OR REPLACE TRIGGER patrocinador_ya_es_colaborador
BEFORE INSERT ON Patrocina
FOR EACH ROW
declare
    cont number(20);
BEGIN
    SELECT count(*)
    into cont
    FROM Colabora
    where email=:new.email
    and numero_edicion=:new.numero_edicion;

    if cont>0 then
        raise_application_error(-
20000, 'El patrocinador a insertar ya esta registrado como colaborador para esta ed
icion.
    Insercion de tupla abortada');
    end if;
END;
/

CREATE OR REPLACE TRIGGER colaborador_ya_es_patrocinador
BEFORE INSERT ON Colabora
FOR EACH ROW
declare
    cont number(20);
BEGIN
    SELECT count(*)
    into cont
    FROM Patrocina
    where email=:new.email

```

```

    and numero_edicion=:new.numero_edicion;

    if cont>0 then
        raise_application_error(-
20000, 'El colaborador a insertar ya esta registrado como patrociandor para esta ed
icion.
        Insercion de tupla abortada');
    end if;
END;
/

```

6.3.2 Subsistema 2

```

CREATE OR REPLACE TRIGGER disparador_s2
BEFORE INSERT ON ocupa
FOR EACH ROW
declare
    cont integer; fecha_aux date;
BEGIN
    SELECT partidos.fecha
    into fecha_aux
    FROM partidos
    WHERE partidos.id_partido=:new.id_partido;
    -- ya tengo la fecha asociada al partido dado

    SELECT count(*)
    into cont
    FROM ocupa NATURAL JOIN partidos
    where id_pista=:new.id_pista AND
           partidos.fecha=fecha_aux ;

    if cont>2 then
        raise_application_error(-
20000, 'Esta pista ya tiene asociados tres partidos. Insercion de tupla abortada');

    end if;
END;
/

```

6.3.3 Subsistema 3

```

CREATE OR REPLACE TRIGGER nuevo_pedido
BEFORE INSERT ON CompuestoPor
FOR EACH ROW
DECLARE
    email1 VARCHAR2(50);
    email2 VARCHAR2(50);
    idMaterial1 VARCHAR2(10);
BEGIN

```

```

SELECT idmaterial
INTO idMaterial1
FROM CompuestoPor
WHERE numerodepedido=:new.numerodepedido AND idmaterial!=:new.idmaterial;

SELECT EMAIL
INTO email2
FROM Suministra
WHERE idmaterial=idMaterial1;

SELECT EMAIL
INTO email1
FROM Suministra
WHERE idmaterial=:new.idmaterial;

IF email1!=email2 then
    raise_application_error (-
20600, 'Los materiales de un pedido deben de ser de un mismo patrocinador. Accion a
bortada');
END IF;
END;
/

```

6.3.4 Subsistema 4

```

CREATE OR REPLACE TRIGGER disparador_S4
BEFORE INSERT ON TrabajadorAsignadoA
FOR EACH ROW
declare
    cont number(20);
BEGIN
    SELECT count(*)
    into cont
    FROM TrabajadorAsignadoA
    where numeropista=:new.numeropista
    and numero=:new.numero
    and dni=:new.dni
    and fecha=:new.fecha;

    if cont>0 then
        raise_application_error(-
20000, 'La asignacion de esa pista a ese trabajador ha fallado ya que ese trabajado
r ya habia sido asignado a esa pista ese dia.
Insercion de tupla abortada');
    end if;
END;
/

```


6.4 Interfaces implementadas

6.4.1 Subsistema 1

```
import java.sql.*;
import java.sql.SQLException.*;
import java.util.Scanner;
import oracle.jdbc.driver.*;

public class Consultas {

    Connection conn;
    Statement stmt;

    public static void main(String[] args) throws Exception{

        int numero_edicion = -1;
        float dinero_aporteado;

        final Scanner in = new Scanner (System.in);

        Consultas consultas = new Consultas();

        consultas.initialize();

        //A partir de aqui depende de la interfaz que tenga cada uno:
        System.out.println("Introduzca numero de edicion: ");

        while(numero_edicion < 0){
            numero_edicion = Integer.parseInt(in.nextLine());

            if (numero_edicion < 0)
                System.out.println("Numero de edicion introducido no valido.
                                   Por favor, introduzcalo de nuevo: ");
        }

        dinero_aporteado = consultas.consultaDineroAporteado(numero_edicion);

        System.out.println("El dinero total que los colaboradores aportaron en la
                           edicion " + numero_edicion + " fue: " + dinero_aporteado + "\n");
    }

    //Solo tocar usuario y contrasena
    public void initialize() throws Exception {
        DriverManager.registerDriver( new oracle.jdbc.driver.OracleDriver());

        //Poner el usuario y contresena:
        conn = DriverManager.getConnection
            ("jdbc:oracle:thin:@oracle0.ugr.es:1521/practbd.oracle0.ugr.es",
            "usuario","contrasena");

        System.out.println("Conexion a la base de datos establecida");
    }

    //Esta es la funcion donde se hace la consulta
    public float consultaDineroAporteado(int numero_edicion) throws Exception {
        float total = 0;

        stmt = conn.createStatement ( );

        ResultSet rset = stmt.executeQuery ("SELECT dinero_aporteado FROM Colabora
                                             where numero_edicion = '" + numero_edicion + "'");

        while (rset.next ()) {
            total += Float.parseFloat(rset.getString(1));
        }

        return total;
    }
}
```

```
}

```

6.4.2 Subsistema 2

```
import java.awt.Container;
import java.awt.event.*;
import java.sql.*;
import oracle.jdbc.*;
import java.util.*;

import java.util.Scanner;

public class ConsultasS2 {

    Connection conn;
    Statement stmt;
    PreparedStatement pstmt;

    /*****
    public static void main(String[] args) throws Exception
    {
        String aux = new String();
        int idPista = -1;
        String nombrePista = new String();
        Scanner in = new Scanner (System.in);

        System.out.println("Introduzca el identificador de pista");
        aux = in.nextLine(); // System.in.read();
        idPista = Integer.parseInt(aux);

        System.out.println("Introduzca capacidad: ");
        aux = in.nextLine();
        int capacidad = Integer.parseInt(aux);

        System.out.println("Introduzca el nombre de la pista");

        nombrePista = in.nextLine() ;
        // alonso ----

        ConsultasS2 consultas = new ConsultasS2();

        consultas.initialize();

        //A partir de aqui depende de la interfaz que tenga cada uno:
        // procesar consulta
        consultas.insertData(1, nombrePista, capacidad);

        // comprobando el INSERT
        int result = consultas.insertData(idPista, nombrePista, capacidad);
        System.out.println("result = " +result);
        if (result != 0)
        {
            System.out.println("ERROR al introducir los campos");
            System.exit(1) ;
        }

        System.out.println("Fila añadida correctamente");

    }

    /*****
    //Solo tocar usuario y contrasena
    public void initialize() throws Exception {
        DriverManager.registerDriver( new OracleDriver()); // oracle.jdbc.driver.
        System.out.println("aqui,...");
        //Poner el usuario y contresena:
        conn = DriverManager.getConnection
        ("jdbc:oracle:thin:@oracle0.ugr.es:1521/practbd.oracle0.ugr.es",
```

```

                                "x6067525", "x6067525");

        System.out.println("Conexion a la base de datos establecida");
    }

    // *****
    public int insertData (int numero, String lastName, int capacidad) throws
Exception
    {
        try
        {
            pstmt = conn.prepareStatement("INSERT INTO pista
(numeropista,nombre,capacidad) VALUES (?, ?, ?)");

            pstmt.setInt(1, numero);
            pstmt.setString(2, lastName);
            pstmt.setInt(3, capacidad);

            pstmt.executeUpdate();

            //conn.commit();

            return 0; // return OK
        }
        catch (SQLException e)
        {
            return 1; // return error
        }
    }
}

```

6.4.3 Subsistema 3

```

import java.awt.Container;
import java.awt.event.*;
import java.sql.*;
import java.sql.SQLException.*;
import java.util.Scanner;
import oracle.jdbc.driver.*;
import java.util.*;
import java.text.SimpleDateFormat;
import java.util.Date;

import java.util.Scanner;

public class Interfaz_S3 {

    Connection conn;
    Statement stmt;
    PreparedStatement pstmt;

    public static void main(String[] args) throws Exception{

        String aux = new String();
        String dniTrabajador = new String();
        int edicion = -1;
        int numeroPedido = -1;

        Scanner in = new Scanner (System.in);

        Interfaz_S3 interfaz = new Interfaz_S3();

        interfaz.initialize();

        //Leemos el dni del trabajador
        System.out.println("Introduzca el dni del trabajador: ");
    }
}

```

```

dniTrabajador = in.nextLine();

//Leemos el numero de la edicion
System.out.println("Introduzca el numero de la edicion: ");
aux = in.nextLine();
edicion = Integer.parseInt(aux);

//Leemos el numero de pedido
System.out.println("Introduzca el numero del pedido: ");
aux = in.nextLine();
numeroPedido = Integer.parseInt(aux);

//Leemos la fecha
System.out.println("Introduzca la fecha de entrega (formato dd/mm/yyyy): ");
String fecha = in.nextLine();
SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
Date testDate = null;
String date = fecha;
try{
    testDate = df.parse(date);
    System.out.println("Se ha registrado la fecha indicada, "+testDate);
} catch (Exception e){System.out.println("invalid format");}

if (!df.format(testDate).equals(date)){
    System.out.println("invalid date!!");
}
else{
    System.out.println("valid date!!");
}

//Insertamos los datos en la base de datos
//A partir de aqui depende de la interfaz que tenga cada uno
interfaz.insertData(numeroPedido, date, edicion, dniTrabajador);

// comprobando el INSERT
int result = interfaz.insertData(numeroPedido, date, edicion, dniTrabajador);
System.out.println("result = " +result);
if (result != 0)
{
    System.out.println("ERROR al introducir los campos");
    System.exit(1) ;
}

System.out.println("Fila insertada correctamente");
}

//Solo tocar usuario y contrasena
public void initialize() throws Exception {
    DriverManager.registerDriver( new oracle.jdbc.driver.OracleDriver());

    //Poner el usuario y contresena:
    conn = DriverManager.getConnection
("jdbc:oracle:thin:@oracle0.ugr.es:1521/practbd.oracle0.ugr.es",
    "x7769091","x7769091");

    System.out.println("Conexion a la base de datos establecida");
}

//Funcion para insertar los datos en la tabla
public int insertData (int numeroPedido, String fecha, int edicion, String
dniTrabajador ) throws Exception
{
    try
    {
        pstmt = conn.prepareStatement("INSERT INTO Entrega
(numerodepedido,fechaentrega,numero,dni) VALUES (?,to_date(?, 'dd/mm/yyyy'),?,?)");
        pstmt.setInt(1, numeroPedido);
    }
}

```

```

        pstmt.setString(2, fecha);
        pstmt.setInt(3, edicion);
        pstmt.setString(4, dniTrabajador);

        pstmt.executeUpdate();

        return 0; // return OK
    }
    catch (SQLException e)
    {
        return 0; // return error
    }
}
}

```

6.4.4 Subsistema 4

```

//Para que funcione tiene que estar en la misma carpeta que este archivo el ojdbc8.jar
//Para compilar: javac -classpath ojdbc8.jar Consultas.java
//Para ejecutar: java -cp .:ojdbc8.jar Consultas

import java.sql.*;
import java.sql.SQLException.*;
import java.util.Scanner;
import oracle.jdbc.driver.*;
import java.util.Date;

public class interfaz_S4 {

    Connection conn;
    Statement stmt;

    public static void main(String[] args) throws Exception{

        int numero_edicion = -1;

        String[] trabajadores_libres = new String[20];

        final Scanner in = new Scanner (System.in);

        interfaz_S4 consultas = new interfaz_S4();

        consultas.initialize();

        //A partir de aqui depende de la interfaz que tenga cada uno:

        //introducir edicion correcta
        System.out.println("Introduzca numero de edicion: ");
        while(numero_edicion < 0){
            numero_edicion = Integer.parseInt(in.nextLine());

            if (numero_edicion < 0)
                System.out.println("Numero de edicion introducido no valido. Por favor,
introduzcalo de nuevo: ");
        }

        trabajadores_libres = consultas.consultaTrabajadoresLibres(numero_edicion);

        System.out.println("Los trabajadores libres para realizar el trabajo son:\n");

        for(int i=0; i< 20 ; i++){
            if (trabajadores_libres[i] != null)
                System.out.println( trabajadores_libres[i] + "\n");
        }
    }
}

```

```

    }

}

/*****
//Solo tocar usuario y contrasena
public void initialize() throws Exception {
    DriverManager.registerDriver( new oracle.jdbc.driver.OracleDriver());

    //Poner el usuario y contresena:
    conn = DriverManager.getConnection
("jdbc:oracle:thin:@oracle0.ugr.es:1521/practbd.oracle0.ugr.es",
    "x3593047","x3593047");

    System.out.println("Conexion a la base de datos establecida");
}

/*****
//Esta es la funcion donde se hace la consulta
public String[] consultaTrabajadoresLibres(int numero_edicion) throws Exception {

    String[] trabajadores_libres = new String[20];
    int j=0;

    stmt = conn.createStatement ( );

    ResultSet rset = stmt.executeQuery
        ("SELECT dni FROM TrabajaEn where numero = " + numero_edicion + "");

    while (rset.next ()) {
        trabajadores_libres[j]=rset.getString(1);
        j++;
    }

    return trabajadores_libres;
}
}

```

En Granada, a 17 de marzo de 2020.