

Tema 3

BASES DE DATOS

Fundamentos de Informática
Grado en Ing. Química



ugr

Universidad
de Granada

Jesús Alcalá y David Pelta



DECSAI
Universidad de Granada

Índice

1. Conceptos básicos.
2. Sistemas gestores de bases de datos.
3. Diseño de bases de datos.
4. Bases de datos relacionales.

Objetivos

- Saber que son y para que sirven las bases de datos.
- Conocer los Sistemas de Gestión de Base de Datos y que ventajas nos proporcionan.
- Realizar un diseño lógico de una base de datos a través del modelo de datos Entidad/Relación
- Definir un diseño implementable de una base de datos utilizando el modelo relacional.

Bibliografía

- C.J. Date: “Introducción a los sistemas de bases de datos”. Prentice Hall, 2001 [7ª edición]. ISBN 968-444-419-2.
- Ramez A. Elmasri & Shamkant B. Navathe: “Fundamentos de Sistemas de Bases de Datos”. Addison-Wesley, 2002 [3ª edición]. ISBN 84-782-9051-6.
- Thomas M. Connolly & Carolyn E. Begg: “Sistemas de Bases de Datos”. Addison-Wesley, 2005 [4a edición]. ISBN 84-782-9075-3.
- Olga Pons Capote; Nicolás Marín Ruiz; Juan Miguel Medina Rodríguez; Silvia Acid Carrillo; Mª Amparo Vila Miranda. Introducción a las bases de datos. El modelo relacional.. Paraninfo. 2005.
- Henry F. Korth, Abraham Silberschatz & S. Sudarshan: “Fundamentos de Bases de Datos”. McGraw-Hill, 2006 [5ª edición]. ISBN 84-481-4644-1.

1. Conceptos básicos

Ejemplo: Gestión de la Universidad

- Utilización de aplicaciones distintas para cada necesidad.
- Cada programa define y gestiona sus propios datos



<i>Aplicaciones</i>	<i>Datos</i>
Matrículas	Alumnos, asignaturas...
Actas	Asignaturas, profesores, alumnos...
Expedientes y certificados	Alumnos, planes de estudio...
Organización docente	Profesores, asignaturas, departamentos...
Horarios	Profesores, grupos, asignaturas, aulas...
...	

1. Conceptos básicos

Ejemplo: Gestión de la Universidad



Problema: Los datos pertenecen a las aplicaciones

- Existencia de datos redundantes
- Posible aparición de inconsistencias
- Información no reutilizable
- Nuevas aplicaciones suponen nuevos datos
- Dificultad a la hora de relacionar datos:

¿En qué aula está un alumno a una hora determinada?

1. Conceptos básicos

Ejemplo: Gestión de la Universidad



Solución: Tener todos los datos almacenados en una base de datos común para todas la aplicaciones

- En la base de datos se almacenan, de forma centralizada, los datos referentes a asignaturas, planes de estudio, cursos, grupos, alumnos, profesores, departamentos, aulas...
- Las distintas aplicaciones trabajan sobre los mismos datos, que se comparten en función de las necesidades de cada aplicación.

1. Conceptos básicos



- **Definición intuitiva:** Una **Base de Datos (BD)** es conjunto de datos comunes relacionados entre sí que se almacenan sin redundancia para ser útiles en diferentes aplicaciones.
- **Definición formal:** Una BD es un fondo común de información almacenada en una computadora relativa a un mismo contexto, organizada de forma sistemática para servir a un propósito específico. La información debe estar disponible para cualquier persona o programa autorizado para acceder a ella.

1. Conceptos básicos



Los datos son hechos conocidos, que pueden registrarse y que tienen un significado implícito.

Ejemplos:

- Nombres
- Números de teléfono
- Direcciones

1. Conceptos básicos

Propiedades:



- Una BD representa algún aspecto del mundo real.
- Toda BD se diseña, construye y carga con datos para un propósito específico.
- Está dirigida a un grupo de usuarios y tienen ciertas aplicaciones preconcebidas que interesan a dichos usuarios.

1. Conceptos básicos



- Las BD pueden tener cualquier tamaño.
- Se pueden generar:
 - Manualmente.
 - Mecánicamente.
- Las BD se pueden crear y mantener con un grupo de programas de aplicación escritos específicamente para esa tarea o mediante un **Sistema de Gestión de BD**.

2. Sistemas Gestores de Base de Datos

Limitaciones e inconvenientes de los sistemas basados en archivos:

- Modificar dinámicamente la estructura de un archivo.
- Consultar o actualizar un archivo de forma flexible.
- Proteger ciertos datos de usuarios no autorizados.
- Permitir el acceso a los datos desde distintas aplicaciones.

Solución: Utilizar un Sistema de Gestión de BD (SGBD)

2. Sistemas Gestores de Base de Datos

- **Definición:** Elemento software con capacidad para definir, mantener y utilizar una base de datos.

- **Ejemplos:**

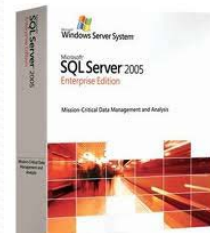
- Oracle



- Access



- ...



- Microsoft SQL Server



- MySQL



- IBM DB2

2. Sistemas Gestores de Base de Datos

Objetivos:

- Los datos se organizan independientemente de las aplicaciones que los vayan a usar (independencia lógica) y de los ficheros en los que vayan a almacenarse (independencia física).
- Los usuarios y las aplicaciones pueden acceder a los datos mediante el uso de lenguajes de consulta (p.ej. SQL, Query-by-example...).
- Los datos se gestionan de forma centralizada e independiente de las aplicaciones.
- Consistencia e integridad de los datos.
- Fiabilidad (protección frente a fallos) y seguridad (no todos los datos deben ser accesibles a todos los usuarios).

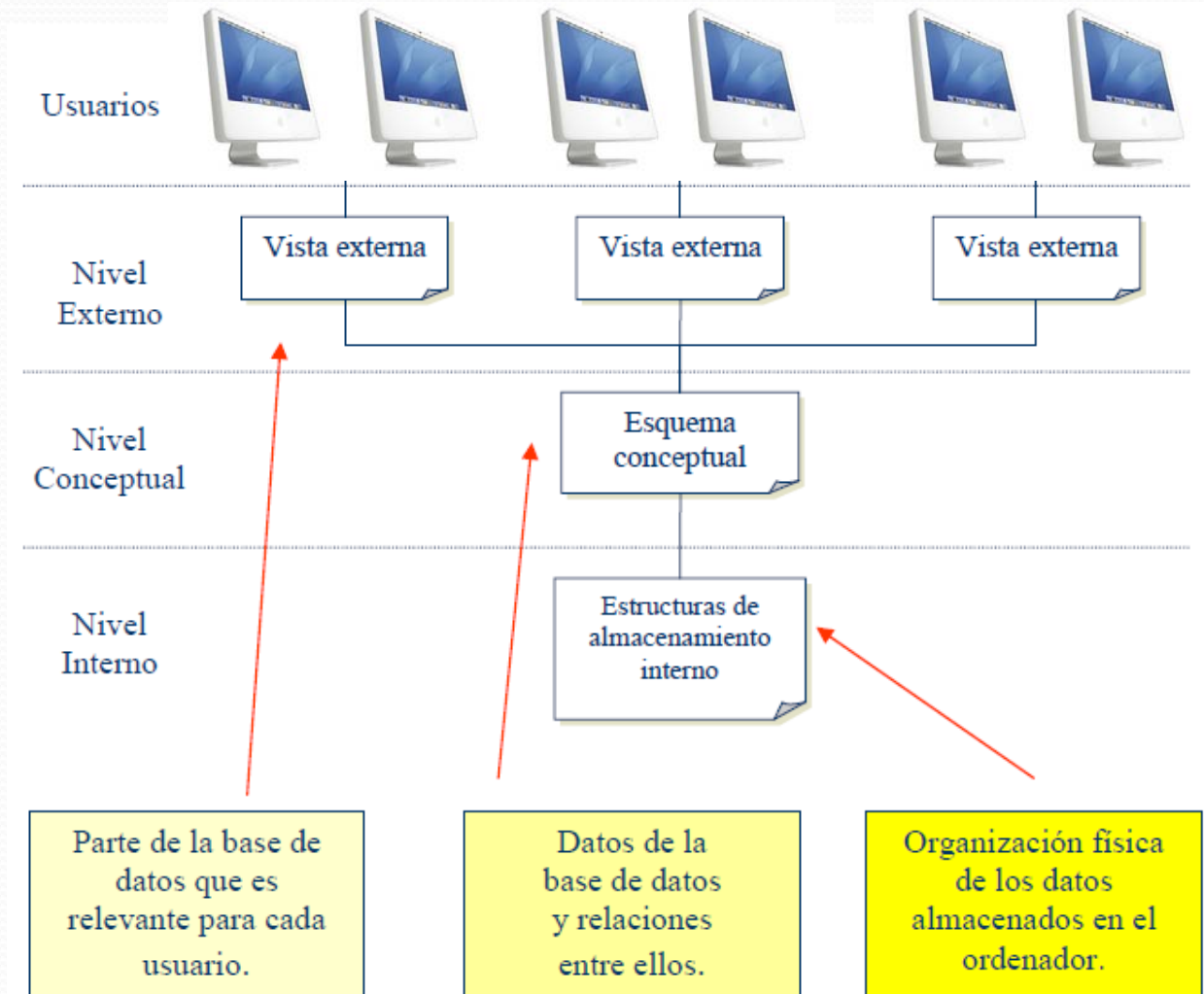
2. Sistemas Gestores de Base de Datos

Ventajas:

<i>Característica</i>	<i>Ventajas</i>
Independencia física	<ul style="list-style-type: none">- Simplifica el desarrollo de las aplicaciones, que no han de preocuparse del almacenamiento de los datos.- Permite realizar cambios en la estructura física para optimizar el uso de la BD sin tener que modificar las aplicaciones.
Independencia lógica	<ul style="list-style-type: none">- Facilita la realización de cambios en las aplicaciones sin tener que modificar la base de datos.- Cada aplicación organiza los datos en función de sus propias necesidades.
Lenguajes de consulta	<ul style="list-style-type: none">- Facilitan al usuario el acceso a los datos.
Centralización	<ul style="list-style-type: none">- Gestión automática de accesos concurrentes.
Consistencia	<ul style="list-style-type: none">- Mantenimiento de la integridad de los datos. p.ej. Datos duplicados, valores incorrectos...
Seguridad	<ul style="list-style-type: none">- Control sobre el acceso a los datos particulares.- El uso de bases de datos permite que los datos se almacenen sin redundancia.

2. Sistemas Gestores de Base de Datos

Arquitectura:



2. Sistemas Gestores de Base de Datos

Arquitectura (cont.):

- **Nivel Interno:** Almacenamiento físico de los datos.
- **Nivel conceptual/lógico:** Descripción de los datos y de sus relaciones → modelo
- **Nivel externo:** Definición de vistas (partes de la BD visibles para las distintas aplicaciones y usuarios).

Objetivo:

Separar las aplicaciones del usuario y la base de datos física

2. Sistemas Gestores de Base de Datos

- **Definición de Esquema:** Descripción de una BD a cualquier nivel interpretable por el SGBD.
- **Esquema interno:** Describe la estructura de almacenamiento de la BD. Campos y ficheros almacenados,....
- **Esquema conceptual:** Describe la estructura lógica global de la BD, la cual debe ser realizada mediante un modelo abstracto de datos comprensible por el SGBD. La descripción debe incluir:
 - Descripción de atributos.
 - Descripción de entidades y conexiones.
 - Restricciones semánticas
- **Esquema externo:** Describe la estructura lógica de la parte de una BD que necesita una o varias aplicaciones concretas. Normalmente se suele utilizar el mismo modelo abstracto que el del esquema conceptual. Pueden haber muchos, incluso uno para cada usuario.

3. Diseño de Base de Datos

Problema de representación de la información

a) **Mundo real:** objetivos, seleccionar datos, hipótesis semánticas, organizar datos.



Biblioteca doméstica.

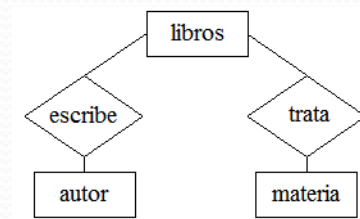
b) **Esquema Inicial:** datos, asociaciones, restricciones.



Libros: título, ISBN, fecha,...

Autor: nombre, nacionalidad,...

Materia: código, descripción,...



c) **Representación lógica:** tablas, ó ...



Libros

Título	ISBN	Fecha	Editorial
--------	------	-------	-----------

Autor

Nombre	Nacio.	Fecha
--------	--------	-------

d) **Esquemas conceptuales y externos.**



Create table libros (...)

...

e) **Representación física:** ficheros,



Fichero de libros

Índice por título de....

3. Diseño de Base de Datos

- Una característica fundamental del enfoque de BD es que proporciona cierto nivel de abstracción de los datos al ocultar detalles de almacenamiento que la mayoría de los usuarios no necesitan conocer.
- Los **modelos de datos** son el principal instrumento para ofrecer dicha abstracción

3. Diseño de Base de Datos

Modelo de datos:

Mecanismo formal para representar y manipular información de manera general y sistemática. Debe constar de:

1. Descripción de datos.
2. Descripción de las operaciones.
3. Descripción de las reglas de integridad.

3. Diseño de Base de Datos

Clasificación por finalidad:

1. Descripción inicial de la información (fase b). Modelos semánticos. Ej., Modelo Entidad/Relación (E/R).
2. Descripción de la información implementable en un lenguaje de alto nivel (fases c y d). Modelos implementables:
 - Relacional
 - Basados en grafos
 - Orientados a objetos
 - Lógicos
3. Descripción de la información para ser almacenada. Modelos físicos (fase e).

3. Diseño de Base de Datos

Modelado Semántico

- Consiste en estudiar los datos que se pretenden almacenar en la base de datos antes de elegir el modelo de datos concreto que se va a usar en la base de datos.
- El modelado semántico permite separar el análisis (¿qué?) del diseño (¿cómo?).
- Nos centramos en el **modelo E/R**

3. Diseño de Base de Datos

Modelado E/R

- Técnica de análisis basada en la identificación de las **entidades** y de **las relaciones** que se dan entre ellas en la parte de realidad que pretendemos modelar.
- El modelo E/R permite representar de forma abstracta los datos que se pretenden almacenar en una base de datos.
- Existen notaciones alternativas para la representación gráfica del diseño conseguido mediante la técnica de análisis que propone el modelo E/R:
 - Diagramas E/R
 - Diagramas UML (Lenguaje Unificado de Modelado)
 - Diagramas CASE*Method

3. Diseño de Base de Datos

Elementos del modelo E/R



- **Entidad:** Objeto, real o abstracto, distinguible de otros objetos.

Al grupo de entidades con cualidades similares acerca de los cuales se almacena información se le **denomina tipo** (o, simplemente, conjunto de entidades).

Ejemplos:

- Libro
- ...



Jesús Alcalá y David Pelta

- Autor



3. Diseño de Base de Datos

Elementos del modelo E/R



- **Atributo:** Propiedad asociada a una entidad.

Mediante los atributos representamos las propiedades de los objetos.

Para cada atributo hay un conjunto de valores permitidos llamado **dominio**.

Ejemplos:

- del libro: título, ISBN, fecha, editorial, ...
- del autor: nombre, apellidos, nacionalidad,...

3. Diseño de Base de Datos

Elementos del modelo E/R



- **Clave:** Conjunto de atributos que permite identificar unívocamente a una entidad dentro de un conjunto de entidades.

Ejemplos:

- del libro: ISBN
- del autor: nombre y apellidos

3. Diseño de Base de Datos

Elementos del modelo E/R



- **Superclave:** Conjunto de atributos que permite identificar unívocamente a una entidad dentro de un conjunto de entidades.
- **Clave candidata:** Superclave con un número mínimo de atributos.
- **Clave primaria:** Clave candidata elegida por el diseñador de la base de datos para identificar unívocamente a las distintas entidades de un tipo.
- **Clave alternativa:** Cualquiera de las claves candidatas no elegidas por el diseñador de la base de datos.

3. Diseño de Base de Datos

Elementos del modelo E/R

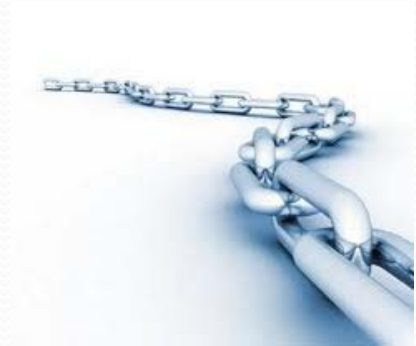


Ejemplos:

- Profesor :
 - {Nombre} no es una clave
 - {NRP} es una clave candidata
 - {NRP, Nombre} es una superclave
 - {Nombre, Despacho} podría serlo (¿o no?)
- Película:
 - {Título} no es una clave
 - {Título, género, año} podría serlo
 - {Título, director, año} es una clave

3. Diseño de Base de Datos

Elementos del modelo E/R



- **Relación (conexión o asociación):** Conexión semántica entre dos conjuntos de entidades.

Ejemplo:

- Relación entre los autores y los libros que han escrito.
- Relación entre los libros y la materia a la que pertenecen.

3. Diseño de Base de Datos

Elementos del modelo E/R

Características de las relaciones



- **Grado:** Número de tipos de entidades que participan en la conexión.
- **Cardinalidad:** Número de elementos de un tipo que se conectan con un elemento de otro (restricción que se observa en el dominio del problema y que controla las ocurrencias de las relaciones).

En el caso de las relaciones binarias (grado 2):

- Relaciones muchos a muchos (n:m)
- Relaciones uno a muchos (1:m)
- Relaciones uno a uno (1:1)

3. Diseño de Base de Datos

Elementos del modelo E/R

Claves de una relación



- Las claves nos permiten diferenciar entre sí las distintas entidades.
- Las claves de las relaciones vienen definidas por las claves de las entidades relacionadas:
 - **Relaciones muchos a muchos (N:M):** La clave primaria será la unión de las claves primarias de las entidades participantes en la relación.
 - **Relaciones uno a muchos (1:N):** La clave primaria de la entidad que interviene en la relación con aridad N.
 - **Relaciones uno a uno (1:1):** Las claves primarias de las entidades participantes son claves candidatas de la relación entre entidades.

3. Diseño de Base de Datos

Ejercicios:

- Identificar las entidades, atributos , claves y relaciones para gestionar la asignación de los alumnos a los grupos de prácticas de Fundamentos de Informática.
- Identificar las entidades, atributos , claves y relaciones para gestionar la entrega de los trabajos realizados por un alumno.
- Identificar las entidades, atributos , claves y relaciones para gestionar la entrega de los trabajos realizados en grupo por los alumnos.

3. Diseño de Base de Datos

Ejemplo: BD de la universidad
Requerimientos (restricciones semánticas)



- Los alumnos se matriculan de varias asignaturas (al menos una).
- Una asignatura puede tener varios alumnos matriculados.
- Todo profesor pertenece a un departamento concreto.
- Todo departamento debe tener un director, que es un profesor.
- Un profesor puede impartir varios grupos de la misma asignatura (o de asignaturas diferentes).
- Un grupo de una asignatura ha de estar impartido, al menos, por un profesor.
- Las clases de cada asignatura se imparten días, horas y aulas determinadas.
- Los atributos de cada entidad son “los habituales”.

3. Diseño de Base de Datos

Ejemplo: BD de la universidad

Identificación de entidades, atributos y relaciones



ugr

Universidad
de Granada

Entidades	Asignatura	Alumno	Profesor	Departamento	Aula	Grupo
Atributos	<u>ID</u> Nombre Créditos Carácter Curso ...	<u>DNI</u> Nombre Dirección E-mail ...	<u>NRP</u> Nombre Categoría Área ...	<u>ID</u> Nombre	<u>ID</u> Capacidad	<u>ID</u> Tipo ...

3. Diseño de Base de Datos

Ejemplo: BD de la universidad

Identificación de entidades, atributos y relaciones



ugr

Universidad
de Granada

Relación	Entidades participantes	Cardinalidad	Atributos
se matricula en	Alumno – Grupo	N:M	Calificación
enseña	Profesor – Grupo	N:M	
impartida en	Asignatura – Grupo	1:N	
asignada a	Aula – Grupo	N:M	Día, hora
pertenece a	Profesor – Departamento	N:1	
dirige	Profesor – Departamento	1:1	

3. Diseño de Base de Datos

Representación gráfica del modelo E/R

Tipo de entidad

Grupo de objetos que tienen las mismas propiedades y que en la organización para la que va a servir la BD tienen una existencia independiente, bien sea física o abstracta.

Notación:



3. Diseño de Base de Datos

Representación gráfica del modelo E/R

Tipo de relación

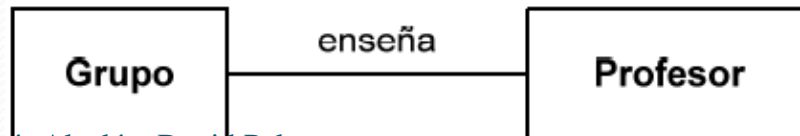
Asociación que se establece entre tipos de entidad para representar un conjunto de relaciones que se establecen entre las ocurrencias de esos tipos de entidades.

Notación:

- E/R clásico



- UML



3. Diseño de Base de Datos

Representación gráfica del modelo E/R

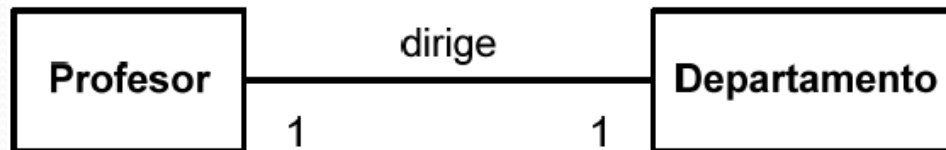
Cardinalidad máxima de una relación

Relación uno a uno

- E/R clásico



- UML



3. Diseño de Base de Datos

Representación gráfica del modelo E/R

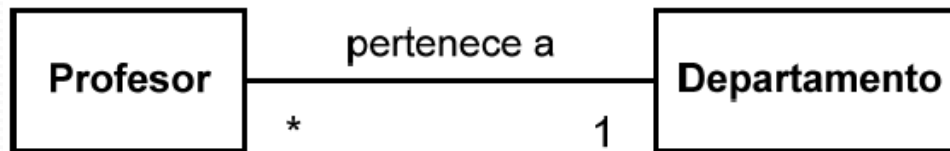
Cardinalidad máxima de una relación

Relación uno a muchos

- E/R clásico



- UML



3. Diseño de Base de Datos

Representación gráfica del modelo E/R

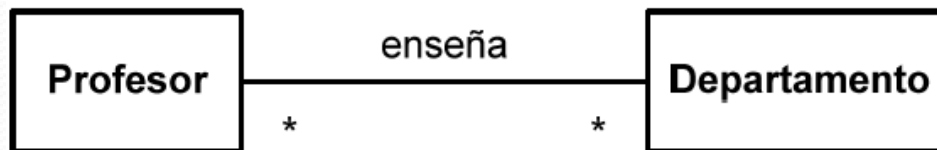
Cardinalidad máxima de una relación

Relación muchos a muchos

- E/R clásico



- UML



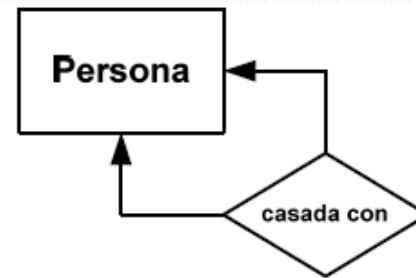
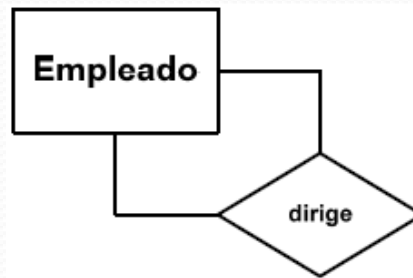
3. Diseño de Base de Datos

Representación gráfica del modelo E/R

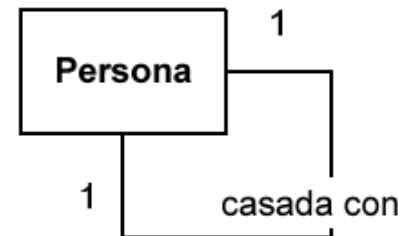
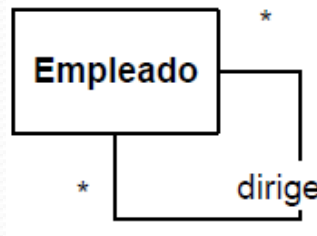
Relación involutiva

Relación de un tipo consigo mismo

- E/R clásico



- UML



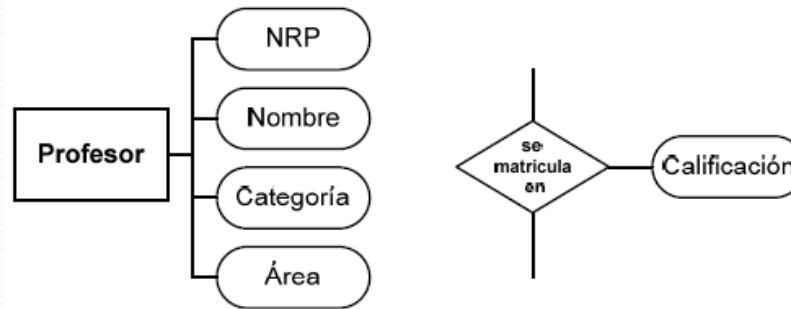
3. Diseño de Base de Datos

Representación gráfica del modelo E/R

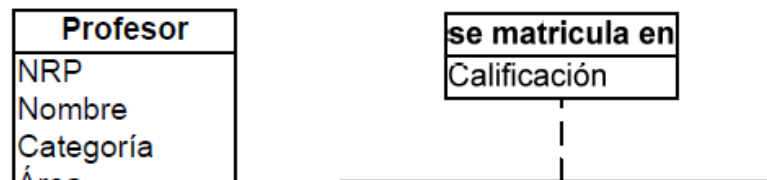
Atributos

Propiedades que caracterizan a las ocurrencias de un tipo de entidad o de un tipo de relación.

- E/R clásico



- UML



3. Diseño de Base de Datos

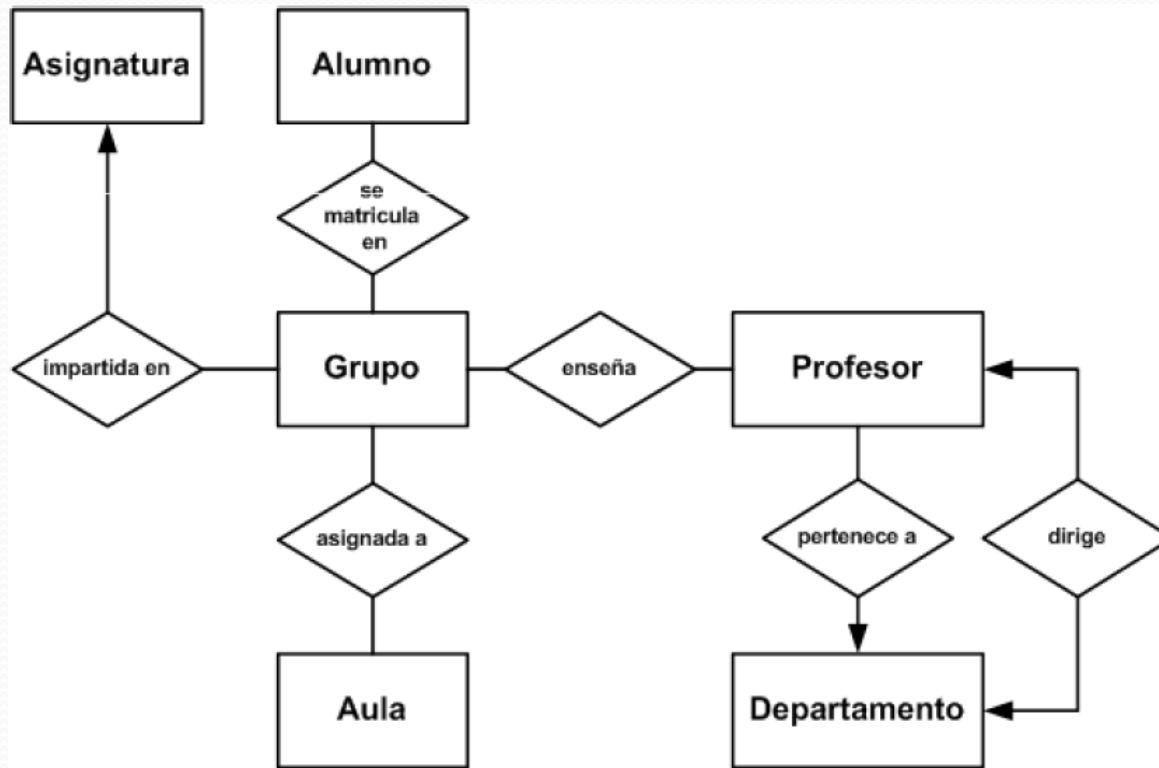
Ejercicios:

- Representación gráfica del modelo para gestionar la asignación de los alumnos a los grupos de prácticas.
- Representación gráfica del modelo para gestionar la entrega de los trabajos realizados por un alumno.
- Representación gráfica del modelo para gestionar la entrega de los trabajos realizados en grupo por los alumnos.

3. Diseño de Base de Datos

Ejemplo: BD de la universidad

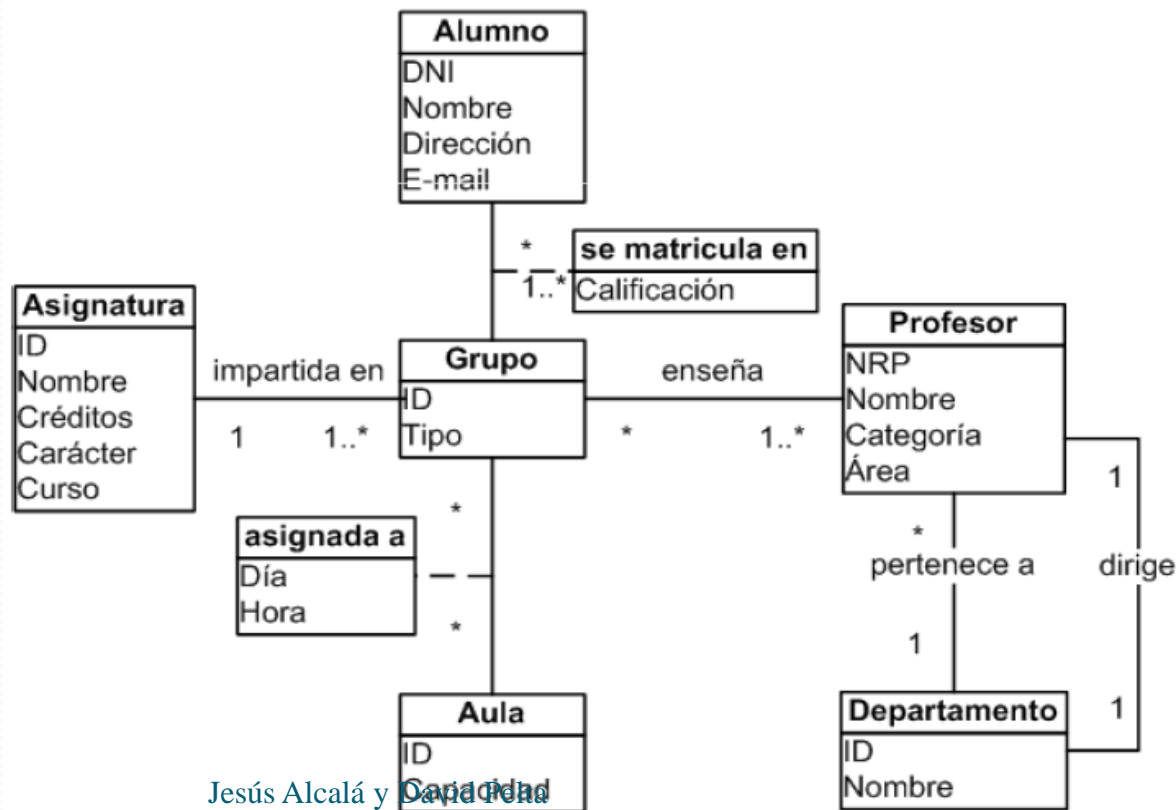
Representación gráfica. Diagrama E/R



3. Diseño de Base de Datos

Ejemplo: BD de la universidad

Representación gráfica. UML



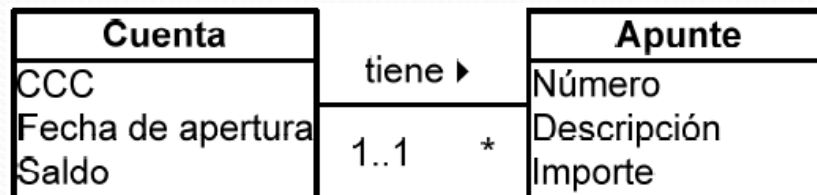
3. Diseño de Base de Datos

Elementos del modelo E/R

Entidades fuertes y entidades débiles

- Un tipo de entidad es **fuerte** si la existencia de sus ocurrencias no depende de ningún otro tipo.
- En caso contrario, se dice que el tipo de entidad es **débil**.

Ejemplo:



Un apunte (entidad débil) sólo puede existir asociado a una cuenta (entidad fuerte).

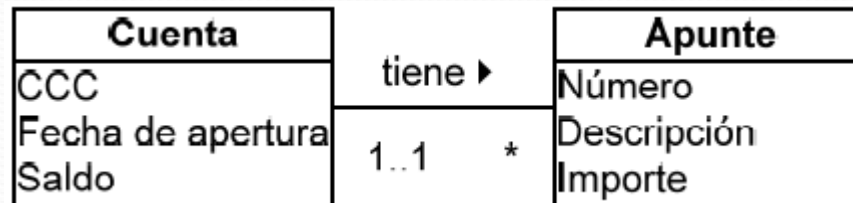
3. Diseño de Base de Datos

Elementos del modelo E/R

Entidades fuertes y entidades débiles

- **Dependencia existencial:** Si se elimina una ocurrencia del tipo de entidad fuerte, habrá que eliminar las ocurrencias del tipo de entidad débil que dependen de ella.

Ejemplo:



Si eliminamos una cuenta, sus apuntes han de desaparecer de la base de datos (si no, tendríamos apuntes que corresponderían a una cuenta que no existe).

3. Diseño de Base de Datos

Elementos del modelo E/R

Entidades fuertes y entidades débiles

- **Claves primarias:** La entidad débil no tiene suficientes atributos propios para formar una clave primaria: La clave primaria de la entidad débil incluye a la clave primaria de la entidad fuerte de la que depende existencialmente.

Clave primaria entidad débil = Clave primaria entidad fuerte + Discriminante

Ejemplo:

{CCC} es la clave primaria de la entidad fuerte “Cuenta”

{CCC, Número} es la clave primaria de la entidad débil “Apunte”

3. Diseño de Base de Datos

Elementos del modelo E/R

Relaciones de especialización y generalización

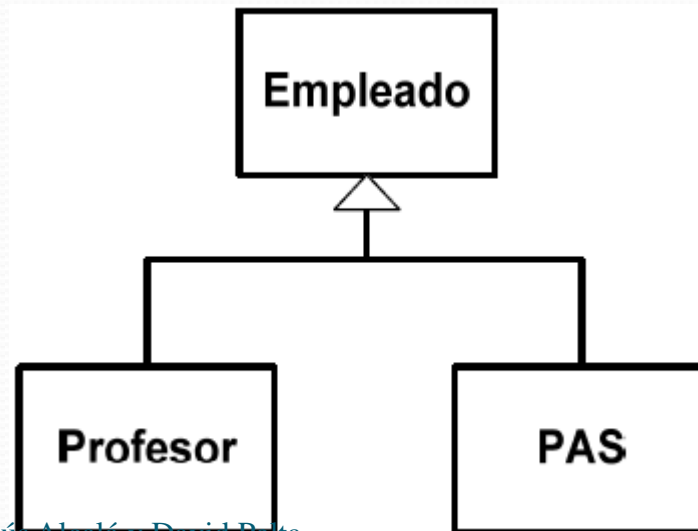
- **Supertipo:** Tipo de entidad que incluye uno o más subgrupos distintos de ocurrencias que deben ser representados en el modelo de datos.
- **Subtipo:** Cada uno de los subgrupos de ocurrencias de un tipo de entidad que se han de representar en el modelo de datos.
- **Especialización:** Proceso de extraer diferencias entre las ocurrencias de un tipo de entidad para distinguir los subtipos que lo forman.
- **Generalización:** Proceso de encontrar la parte común de las ocurrencias de distintos tipos de entidad para extraer el supertipo que los engloba.

3. Diseño de Base de Datos

Elementos del modelo E/R

Relaciones de especialización. Relaciones ES-UN

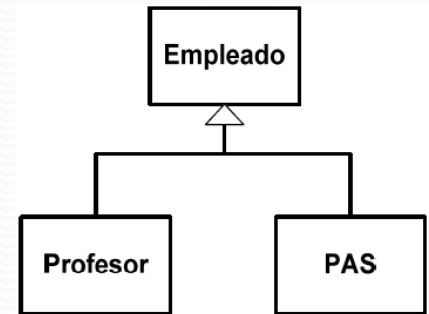
Relaciones que se establecen en un diagrama E/R entre un supertipo y sus subtipos:



3. Diseño de Base de Datos

Elementos del modelo E/R

Relaciones de especialización. Relaciones ES-UN

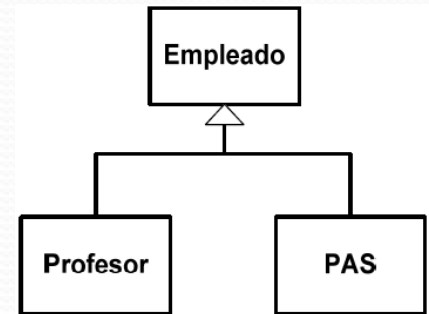


- Los subtipos **heredan** los atributos de los supertipos: Los subtipos poseen todos los atributos del supertipo más algunos propios.
- La clave primaria de los subtipos es la **clave primaria del supertipo**.

3. Diseño de Base de Datos

Elementos del modelo E/R

Relaciones de especialización. Restricciones






- **Participación:** Determina si un miembro de la superclase debe obligatoriamente ser un miembro de una subclase.
- **Exclusividad:** Determina si un miembro de una subclase puede ser, a la vez, miembro de otras subclases.

4. Bases de Datos Relacionales

El modelo de datos relacional organiza y representa los datos en forma de tablas o relaciones:

Una base de datos relacional es una colección de relaciones [tablas].

Representación lógica	Representación física	Modelo relacional
 Tabla	Archivo secuencial	Relación
 Fila	Registro	Tupla
 Columna	Campo	Atributo

4. Bases de Datos Relacionales

El concepto de relación: Tuplas, atributos y dominios

id_trabajador	nombre	tarifa_hr	tipo_de_oficio	id_supv
1235	F. Aguilera	12,50	Electricista	1311
1412	A. Calvo	13,75	Fontanero	1540
2920	N. Marín	10,00	Carpintero	null
3231	O. Pons	17,40	Albañil	null
1540	J.M. Medina	11,75	Fontanero	null
1311	J.C. Cubero	15,50	Electricista	null
3001	D. Sánchez	8,20	Albañil	3231

4. Bases de Datos Relacionales

El concepto de relación: Tuplas, atributos y dominios

- **Atributo (A_i):** Elemento susceptible de tomar valores (cada una de las columnas de la tabla).
- **Dominio (D_i):** Conjunto de valores que puede tomar un atributo (se considera finito).
- **Tupla:** Cada uno de los elementos que contiene una instancia de la relación (filas).

4. Bases de Datos Relacionales

El concepto de relación

Relación $R(A_1..A_n)$

Subconjunto del producto cartesiano $D_1 \times \dots \times D_n$
(esto es, una tabla).

En una relación hay que distinguir dos aspectos:

- **Esquema de la relación:** Los atributos $A_1..A_n$. Por ejemplo:
Trabajadores (id_trabajador, nombre, tarifa_hr, tipo_de_oficio, id_supv)
- **Instancia de la relación:** El conjunto de tuplas $\{(X_1, X_2, \dots, X_n)\} \subseteq D_1 \times D_2 \times \dots \times D_n$ que la componen en cada momento.

4. Bases de Datos Relacionales

El concepto de relación

Relación $R(A_1..A_n)$

Subconjunto del producto cartesiano $D_1 \times \dots \times D_n$
(esto es, una tabla).

Consecuencias de la definición de relación como conjunto de tuplas:

- No existen tuplas duplicadas (concepto de clave primaria).
- No existe orden en las tuplas (ni en los atributos).

4. Bases de Datos Relacionales

Esquema de la BD

Una base de datos relacional es un conjunto finito de relaciones junto con una serie de restricciones o reglas de integridad:

- **Restricción de integridad:** Condición necesaria para preservar la corrección semántica de la base de datos.
- **Esquema de la base de datos:** Colección de esquemas de relaciones junto con las restricciones de integridad que se definen sobre las relaciones.

4. Bases de Datos Relacionales

Instancia de la BD



- **Instancia (o estado) de la base de datos:** Colección de instancias de relaciones que verifican las restricciones de integridad.
- **Base de datos relacional:** Instancia de la base de datos junto con su esquema.

4. Bases de Datos Relacionales

Restricciones de integridad

Asociadas a las tuplas de una relación

$$0 \leq \text{edad} \leq 120$$

$$\text{impuestos} \leq \text{sueldo}$$

En ocasiones, no se conoce el valor de un atributo para una determinada tupla. En esos casos, a ese atributo de esa tupla se le asigna un valor nulo (null), que indica que el valor de ese atributo es desconocido o, simplemente, que ese atributo no es aplicable a esa tupla.

4. Bases de Datos Relacionales

Restricciones de integridad

Asociadas a las tuplas de una relación

- **Clave primaria:** Conjunto de atributos seleccionados para identificar unívocamente a las tuplas de una relación.
- **Integridad de entidad:** Los atributos que componen la clave primaria no pueden tomar valores nulos, ya que la clave primaria debe permitirnos identificar unívocamente cada tupla de la relación.

4. Bases de Datos Relacionales

Restricciones de integridad

Asociadas a las relaciones de la BD

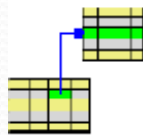
- **Clave externa:** Conjunto de atributos de una relación cuyos valores en las tuplas deben coincidir con valores de la clave primaria de las tuplas de otra relación.
- **Integridad referencial:** Todos los valores no nulos de una clave externa referencian valores reales de la clave referenciada.

4. Bases de Datos Relacionales

Restricciones de integridad

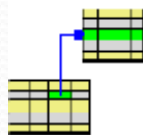
Asociadas a las relaciones de la BD

La integridad referencial mantiene las conexiones en las bases de datos relacionales:



$\text{imparte.NRP} \in \text{profesor.NRP}$

El profesor que imparte una asignatura debe existir en la tabla de profesores.



$\text{cuenta.sucursal} \in \text{sucursal.número}$

Una cuenta tiene que pertenecer a una sucursal existente.

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:

Transformación de un diagrama E/R en un esquema relacional (esto es, en un conjunto de tablas):

1. Se transforman en tablas todos los tipos de entidades y relaciones que aparecen en el diagrama E/R.
2. Se seleccionan las claves primarias para cada una de las tablas de nuestro esquema lógico.
3. Se fusionan aquellas tablas que compartan su clave primaria.

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:
Entidades



Cada tipo de entidad da lugar a una tabla en la base de datos.

- **Atributos:** los atributos del tipo de entidad.
- **Clave primaria:** una de las claves candidatas del conjunto de entidades.

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:
Entidades débiles



Cada tipo de entidad da lugar a una tabla en la base de datos.

- **Atributos:** Además de los atributos propios de la entidad débil, los atributos pertenecientes a la clave primaria de la entidad fuerte de la que depende existencialmente la entidad débil.
- **Clave primaria:** La clave primaria de la entidad fuerte más un conjunto de atributos propio de la entidad débil (discriminante)

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:
Relaciones



Cada tipo de relación da lugar a una tabla en la base de datos.

- **Atributos:** Los atributos de las claves primarias de las entidades que intervienen en la relación más los atributos propios de la relación.

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:
Relaciones.



Clave primaria: Si la relación no tiene atributos propios:

- **Relación muchos a muchos:** La unión de las claves de los conjuntos de entidades que intervienen.
- **Relación uno a muchos:** La clave correspondiente al conjunto de entidades que participa en la relación con cardinalidad “muchos”.
- **Relación uno a uno:** Una de las claves de las entidades intervinientes en la relación (cualquiera).

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:
Relaciones.



Clave primaria: Si la relación tiene atributos propios, los atributos correspondientes al tipo de relación, a los que tal vez añadiremos algunos atributos propios dependiendo de la semántica del problema.

Claves externas: Una por cada una de las claves primarias de las entidades que intervienen en la relación.

Nota: Las relaciones entre entidades débiles y fuertes no hay que pasarlas a tablas porque la relación se recoge como parte de la clave primaria de la entidad débil (la parte correspondiente a la clave primaria de la entidad fuerte es una clave externa que apunta a la tabla derivada de la entidad fuerte).

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:

Fusión de tablas

Se pueden combinar en una sola todas las tablas que comparten su clave primaria.

Ejemplo:

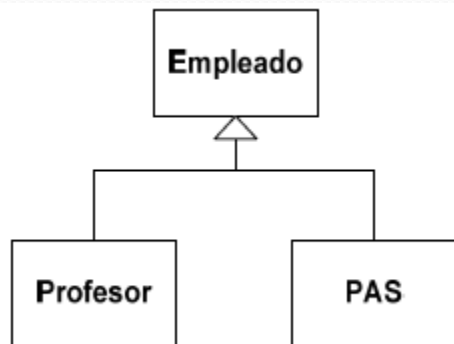
Las tablas derivadas de las relaciones muchos a uno se fusionan con las derivadas de las entidades que participan en la relación con cardinalidad N.

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:

Relaciones de generalización y especialización

- **Estrategia A:** Una tabla por cada conjunto de entidades. Las particularizaciones heredan la clave primaria del conjunto de entidades de nivel superior (la cual será, en las tablas correspondientes a los subtipos, una clave externa que referencia a la tabla derivada del supertipo).



Ejemplo:

Empleado (NRP, nombre, dirección...)

Profesor (NRP, departamento, categoría)

PAS (NRP, grupo, nivel)

4. Bases de Datos Relacionales

Del modelo E/R al modelo relacional:

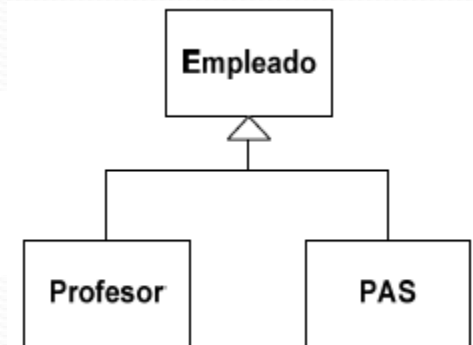
Relaciones de generalización y especialización

- **Estrategia B:** Una tabla por cada caso particular. Las particularizaciones heredan todos los atributos de la entidad general.

Ejemplo:

Profesor (NRP, nombre, dirección... departamento, categoría)

PAS (NRP, nombre, dirección... grupo, nivel)



Ejemplo



Código	Nombre	Ciudad
S1	Juan Gómez	Alicante
S2	Raquel Jiménez	Alicante
S3	Pablo Fernández	Madrid
S4	Carmen López	Granada
S5	Cristina García	Madrid

Código	Nombre	Dimensiones	Peso
P1	Tuerca	12 mm	1.2 gr
P2	Tornillo	10 mm	1.7 gr
P3	Tornillo	20 mm	2.7 gr
P4	Clavo	25 mm	1.4 gr
P5	Arandela	15 mm	3.0 gr
P6	Tuerca	10 mm	1.0 gr

CódigoProveedor	CódigoPieza	Cantidad
S1	P1	300
S1	P2	200
S1	P3	400
S1	P4	200
S1	P5	100
S1	P6	100
S2	P1	200
S2	P2	400
S3	P2	200
S4	P2	200
S4	P4	300
S4	P5	400

4. Bases de Datos Relacionales

Ejercicios:

- Pasar al modelo relacional el modelo para gestionar la asignación de los alumnos a los grupos de prácticas.
- Pasar al modelo relacional el modelo para gestionar la entrega de los trabajos realizados por un alumno.
- Pasar al modelo relacional el modelo para gestionar la entrega de los trabajos realizados en grupo por los alumnos.

Creditos

- Agradecimientos al material proporcionado por Fernando Berzal.