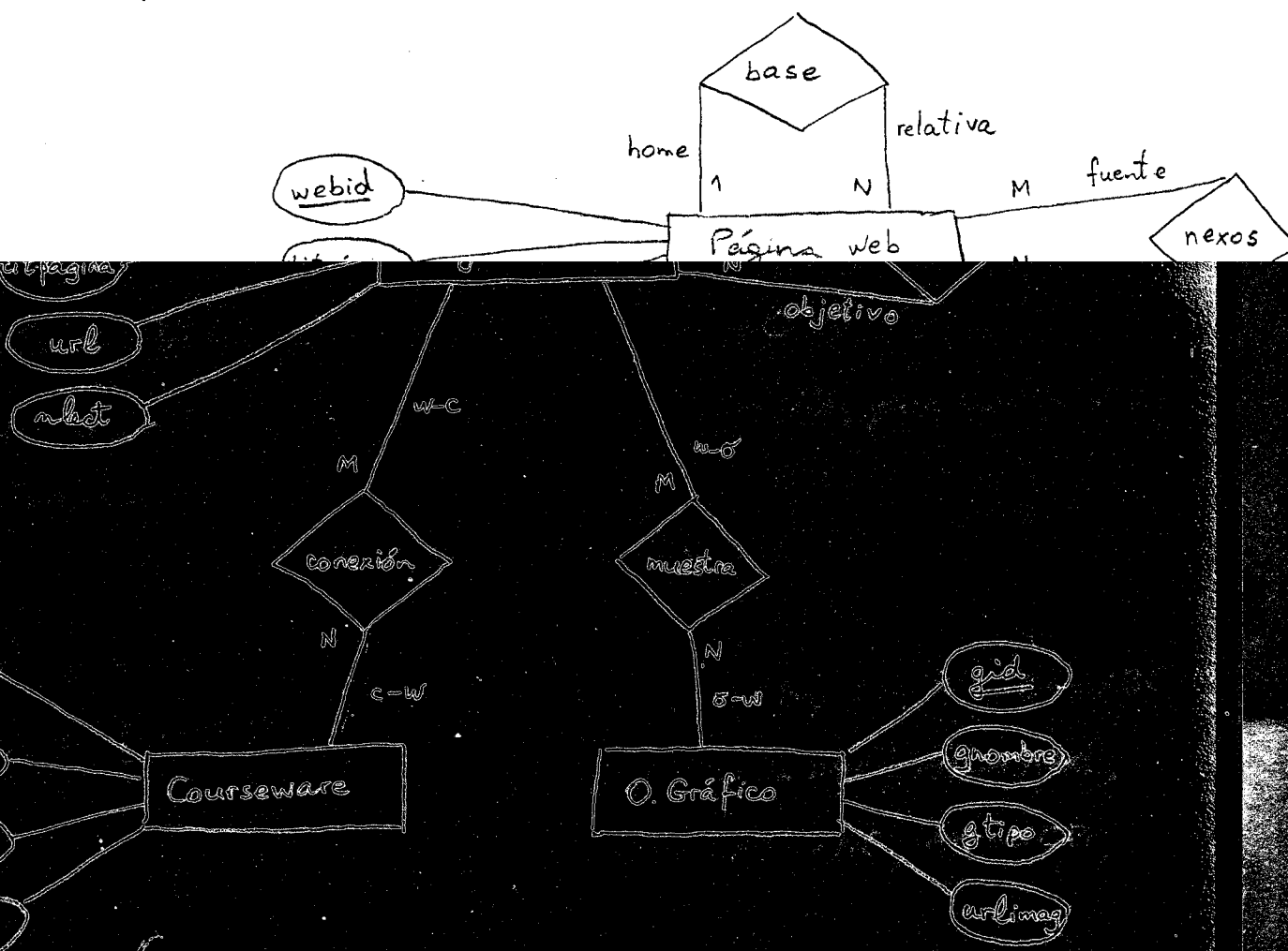


1.



Aspectos secundarios que no deben castigarse si no están presentes en más de 0.1 puntos. Por ej.: nombres para las relaciones, atributos secundarios.

b) Los aspectos importantes son la resolución del modelamiento de página base vs. otras páginas, modelamiento de los nexos (links) arbitrarios entre pág., asociaciones página-courseware, página-objeto gráfico.

1. a) La cerradura se obtiene por aplicación repetida de las reglas de inferencia especificadas por los axiomas de Armstrong:

Usando reflexividad (si $X \supseteq Y$, entonces, $X \rightarrow Y$), obtenemos

$$\begin{aligned} A &\rightarrow AB, A \rightarrow AC, A \rightarrow ABC \\ B &\rightarrow AB, B \rightarrow BC, B \rightarrow ABC \\ C &\rightarrow AC, C \rightarrow BC, C \rightarrow ABC \\ AB &\rightarrow ABC, AC \rightarrow ABC, BC \rightarrow ABC \end{aligned}$$

Usando transitividad, obtenemos $A \rightarrow C$

De aumento (si $X \rightarrow Y$, entonces $XZ \rightarrow YZ$ para cualquier Z), obtenemos:

$$AC \rightarrow BC, AB \rightarrow AC, AB \rightarrow CB$$

La cerradura, entonces, es:

$$D^+ = \{ A \rightarrow AB, A \rightarrow AC, A \rightarrow ABC, B \rightarrow AB, B \rightarrow BC, B \rightarrow ABC, \\ C \rightarrow AC, C \rightarrow BC, C \rightarrow ABC, AB \rightarrow ABC, AC \rightarrow ABC, \\ BC \rightarrow ABC, A \rightarrow C, AC \rightarrow BC, AB \rightarrow AC, AB \rightarrow CB, \\ A \rightarrow B, B \rightarrow C \}$$

Criterio: - si en D^+ se olvida incluir $A \rightarrow B, B \rightarrow C$, restar 0.4

- si en D^+ no se incluye $A \rightarrow C$, restar 1.2

- si en D^+ no se incluyen las df generadas por reflexividad, restar 1.8 puntos

- si en D^+ no se incluyen las df generadas por aumento, restar 1.8 puntos

- si se omite la mención de los teoremas de Armstrong, restar 0.2

- si se omite nombre de cada regla: 0.2 cada una.

.. b)

$$df1: AB \rightarrow C$$

$$df2: C \rightarrow D$$

En este caso, la dependencia $df1$ es total, y también D depende (transitivamente) en forma total de AB . Por lo tanto, S está en 2FN.

Por otro lado, hay una dependencia transitiva implícita ($AB \rightarrow D$), y por lo tanto, S no está en 3FN. //

Criterios: - también está correcto $df1: AB \rightarrow D$
 $df2: D \rightarrow C$

- si no hay justificación alguna, restar 3.0 puntos
- justificaciones parciales o parcialmente incorrectas, restar proporcionalmente

- a) La relación EMPLEADOS está en FNBC: hay una sola df rut \rightarrow nombre, sueldo, tipocavión y no hay dependencias transitivas (con la información provista).

La relación VUELOS, en cambio, está en 2FN, ya que autonomía depende de codavía, y a su vez, este atributo depende de codvuelo (es decir, hay transitividad). Por otro lado, hay una sola clave candidata y es simple (por lo tanto, está en 2FN). (Hay otra df transitiva también, mencionada abajo).

- b) EMPLEADOS1(rut, nombre, sueldo)

PILOTOS(rut, tipocavión)

VUELOS1(codvuelo, desde, hacia, horapartida, horallegada, codavía)

AS(desde, hacia, distancia)

DISTANCIA

(codavía, autonomía)

AVIONES

es preferible porque:

FNBC: no hay dependencias transitivas, y la def. de FNBC

reduce el número de valores nulos que habría EMPLEADOS (criterio informal de diseño).

Este schema

- está en FN

se cumple

- PILOTOS

en EMPLE

- DISTANCIAS elimina la df transitiva que existía en VUELOS referida a las distancias (obviamente, también reduce valores repetidos, como criterio informal).
- AVIONES: idem con respecto a autonomía

Criterios de evaluación parte b):

- reducir 2 puntos por c/a de PILOTOS, DISTANCIAS, AVIONES, si no se incluyen.
- reducir 0.7 puntos si no se da explicación alguna en caso de incluir una de estas nuevas relaciones.

t. a)

CREATE VIEW Acme

AS SELECT código, descripción

FROM Puzas Z, Proveedores P, Catálogo C

WHERE (Z.código = C.código

AND P.rut = C.rut

AND P.nombre = "Ferretería ACME")

AND NOT EXISTS (SELECT *

FROM Proveedores E, Catálogo G

WHERE (Z.código = G.código

AND E.rut = G.rut

AND E.nombre <> "Ferretería ACME")

Criterios de corrección:

- SQL permite varias maneras de especificar una consulta. Si la respuesta es aceptable ^{por SQL} naturalmente, está correcta.
- Ignorar detalles sintácticos tales como apóstrofes o comillas, punto y coma al final, "=" en vez de "<>", etc.
- si no sabe vista, pero escribe una consulta correcta (con SELECT solamente): bajar 1 punto.

4.6)

7

```
SELECT descripción  
FROM Piezas
```

```
WHERE código IN (SELECT Z.código  
FROM Piezas Z, Proveedores P, Catálogo G  
WHERE Z.código = G.código  
AND G.rut = P.rut  
AND P.dirección LIKE '%Providencia%')
```

```
AND código IN (SELECT E.código  
FROM Piezas E, Proveedores V, Catálogo C  
WHERE E.código = C.código  
AND C.rut = V.rut  
AND V.dirección LIKE '%Pudahuel%')
```

```
ORDERED BY descripción
```