



Introducció

La Universitat Politècnica de Wisconsin (UPW) gestiona, a través del seu servei TIC, la compra de les llicències de programari per a tota la comunitat universitària. Habitualment els costos els assumia la mateixa Universitat, i oferia les llicències als centres, departaments i instituts sense cost per a aquestes entitats. L'augment dels costos i el mal ús de les llicències per part d'alguns usuaris han portat a una situació en què cal establir un sistema de copagament de les llicències per part dels centres, departaments o instituts que n'hagen de fer ús.

Per a gestionar la sollicitud de llicències per part de les entitats de la UPW, cal desenvolupar una aplicació informàtica en la intranet, anomenada *Botiga de Programari*, a la qual entren els gestors de les entitats per a seleccionar el tipus i el nombre de llicències que necessiten.

Funcionalitats de l'aplicació

L'aplicació ha de permetre que els gestors hi entren i seleccionen les llicències, que estan organitzades segons la categoria que s'hi ha assignat.

Les llicències disponibles poden ser de dos tipus: amb control d'estoc disponible o sense control d'estoc (cosa que significa que són llicències il·limitades). Per a cada llicència s'indica el preu corresponent i una imatge.

Les categories de les llicències s'organitzen de manera jeràrquica multinivell, indicant en cada categoria quina és la categoria pare.

En les comandes que facen els gestors, hauran d'indicar-hi el centre de cost al qual es carregarà l'import de les llicències i el conjunt de llicències sol·licitades, que compondran les línies de la comanda.

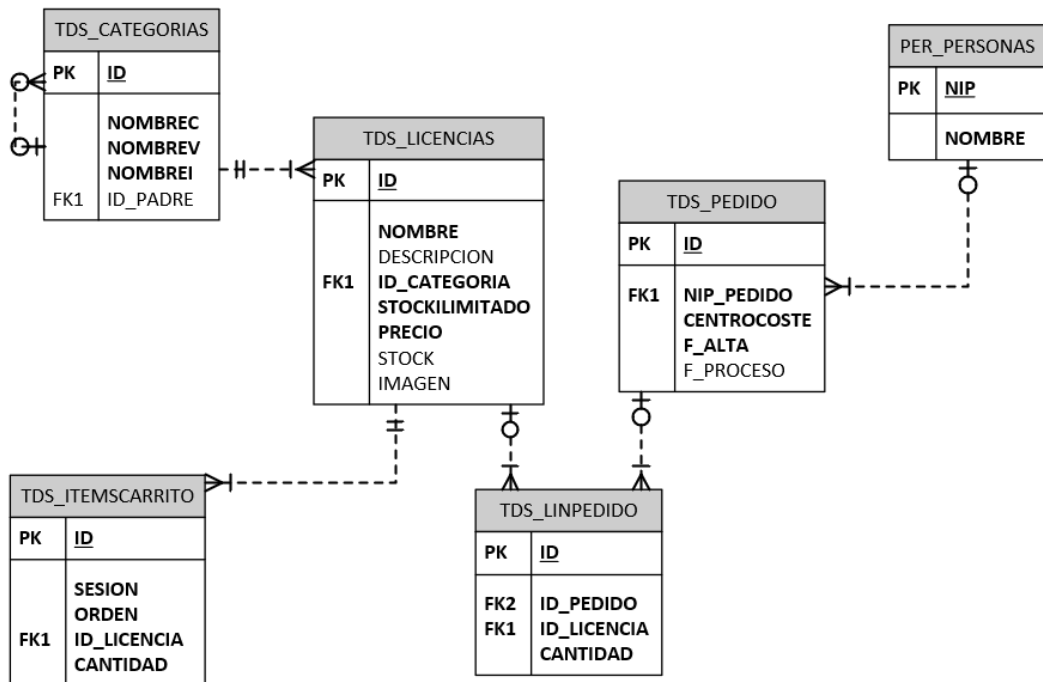
Per a identificar els gestors s'usa la taula de PER_PERSONES de la universitat, on tots els usuaris ja estan donats d'alta.

El funcionament de l'aplicació serà com el d'una botiga en línia: l'usuari, una vegada que s'haja identificat, podrà anar col·locant en el seu carret les llicències que vol adquirir; quan vulga acabar la compra, se li mostrarà el carret, i després que haja emplenat la dada del centre de cost, se li generarà la comanda corresponent.



Diagrama de taules

L'aplicació ja té les taules definides en Oracle 19c, i són les següents:



Llegenda:

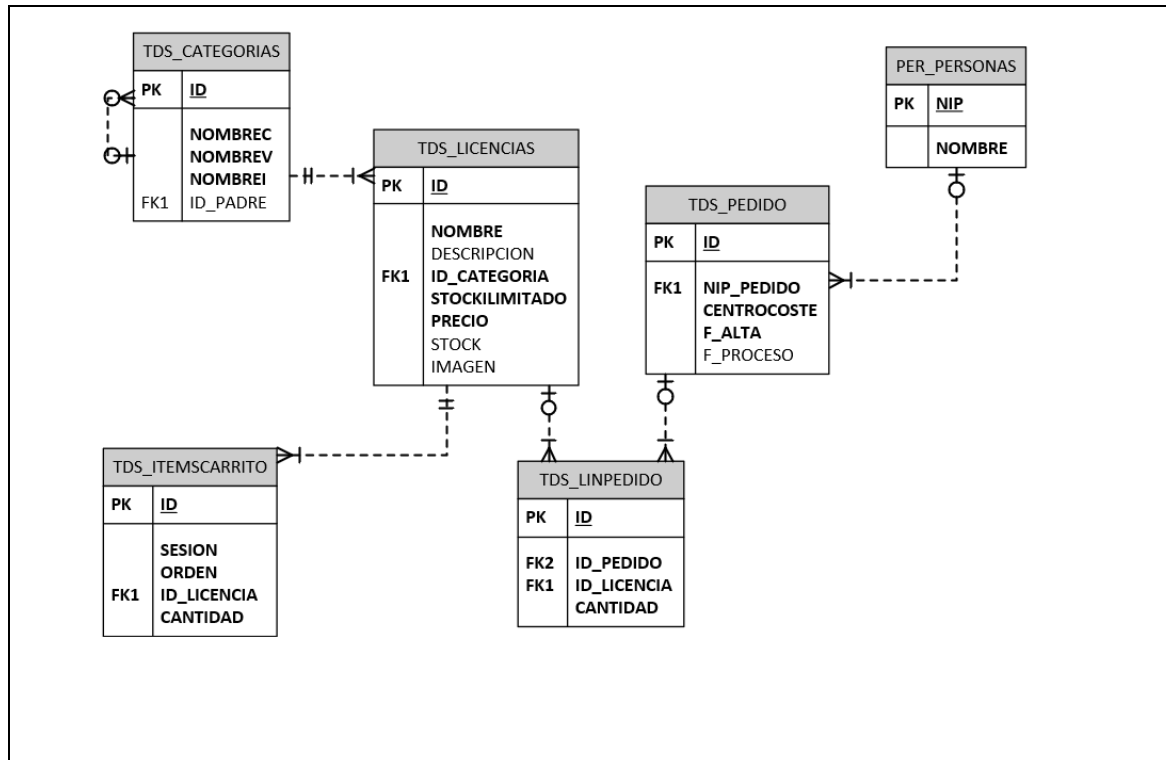
Els camps **no nuls** estan marcats en **negreta**.

FK: claus alienes o *foreign keys*

PK: clau primària o *primary key*

Pregunta 2 (5 punts): Es vol identificar les persones que podran usar les llicències comprades pel gestor de l'entitat. Per a fer-ho, cal modificar l'estructura de dades de manera que es puguin identificar les persones per a les quals s'adquireixen llicències en cada línia de comanda. Completeu en el diagrama la nova taula amb els camps i les relacions que calguen, tenint en compte que totes les persones ja figuren en la taula PER_PERSONAS. Aquesta taula s'ha d'usar posteriorment per a registrar la data d'activació i de caducitat de cada llicència.

Resposta 2:





Pregunta 3 (5 punts): L'estoc s'ha d'actualitzar automàticament quan es cree la comanda. Per a resoldre aquesta funcionalitat cal crear un trigger sobre la taula de línies de comanda que actualitzarà l'estoc de llicències. Cal d'assumir que no es preveuen actualitzacions ni esborraments de línies de comanda.

Resposta 3:

```
CREATE OR REPLACE TRIGGER TDS_LINPEDIDO_TR1
```

```
AFTER ..... ON .....
```

```
FOR EACH ROW
```

```
DECLARE
```

```
BEGIN
```

```
END
```



Supòsit 2

Pregunta 4 (4 punts): Es vol implementar un servei REST dins de l'aplicació que permeta fer operacions en la base de dades relatives al carret (TDS_ITEMSCARRITO) cada vegada que l'usuari afeg un ítem al carret o bé n'elimina un ítem.

Resposta 4a: Completeu el quadre següent, d'especificació del servei per a afegir un ítem al carret:

Mètode HTTP	
Justificació del mètode HTTP que s'ha emprat	
Ruta (<i>endpoint</i>) amb versionament	
Exemple del cos (<i>body</i>) de la sol·licitud en format JSON	
Codi HTTP retornat pel REST en cas que l'operació s'haja fet correctament	
Exemple del cos de la resposta (HTTP CODE i <i>response</i>) en format JSON en cas que l'operació s'haja fet correctament	



Resposta 4b: Completeu el quadre següent, d'especificació del servei per a eliminar un ítem del carret:

Mètode HTTP	
Justificació del mètode HTTP que s'ha emprat	
Ruta (<i>endpoint</i>) amb versionament	
Exemple de crida REST	
Codi HTTP retornat pel REST en cas que l'operació s'haja fet correctament	
Exemple del cos de la resposta (HTTP CODE i <i>response</i>) en format JSON en cas que l'operació s'haja fet correctament	



Pregunta 5 (5 punts): A partir del diagrama entitat-relació de l'aplicació, completeu les anotacions JPA de les classes Java d'entitat següents.

Com a requisit de lògica de negoci, implementeu el mètode d'instància public double calcularImporteTotal() dins de la classe TdsPedido. Aquest mètode ha de calcular i retornar el cost total de la comanda.

Resposta 5a: Completeu la classe TdsLicencias:

```
@
@
public class TdsLicencia {

    @
    @
    private Long id;

    @
    private String nombre;

    @
    private double precio;

    @
    private Integer stock;

    @
    @
    private TdsCategorias categoria;

    public TdsLicencia() {}

    public TdsLicencia(Long id, double precio) {
        this.id = id;
        this.precio = precio;
    }
    /* Getters y Setters se presuponen NO ES NECESARIO COMPLETARLO*/
}
```



Resposta 5b: Completeu la classe TdsPedido:

```
@
@
public class TdsPedido {

    @
    @
    private Long id;

    @
    @
    private TdsPersonas persona;

    @
    private String centroCoste;

    @
    private Date fechaAlta;

    @
    private Date fechaProceso;

    @
    private List<TdsLinPedido> lineasPedido;

    public TdsPedido() {}

    public TdsPedido(Long id, List<TdsLinPedido> lineasPedido) {
        this.id = id;
        this.lineasPedido = lineasPedido;
    }

    // =====
    // MÉTODO DE CÁLCULO DEL IMPORTE TOTAL DEL PEDIDO
    // =====

    /**
     * Calcula el importe total del pedido.
     * @return El importe total del pedido como un valor double.
     */
    public double calcularImporteTotal() {

    }

    /* Getters y Setters se presuponen NO ES NECESARIO COMPLETARLO */
}
}
```



Pregunta 6 (3 punts): Es vol crear la pantalla de compra de llicències mitjançant un carret de la compra. Per a fer-ho s'ha creat una pantalla amb el component CARDS d'APEX perquè se sembla tant com siga possible a altres botigues en línia.

Tenint en compte que les dades de la taula TDS_LICENCIAS són les següents:

ID	NOMBRE	ID_CATEGORIA	PRECIO	STOCK	IMAGEN
1	LICENCIA WORD 2025		299,9	450	(BLOB)
2	LICENCIA EXCEL 2020		123	100	(BLOB)
3	LICENCIA ADOBE 2022		450	200	(BLOB)
4	LICENCIA WORD 2023		100	25	(BLOB)
5	LICENCIA WORD 2020		25	10	(BLOB)
6	LICENCIA EXCEL 2025		300	125	(BLOB)
7	LICENCIA ADOBE 2025		790	350	(BLOB)

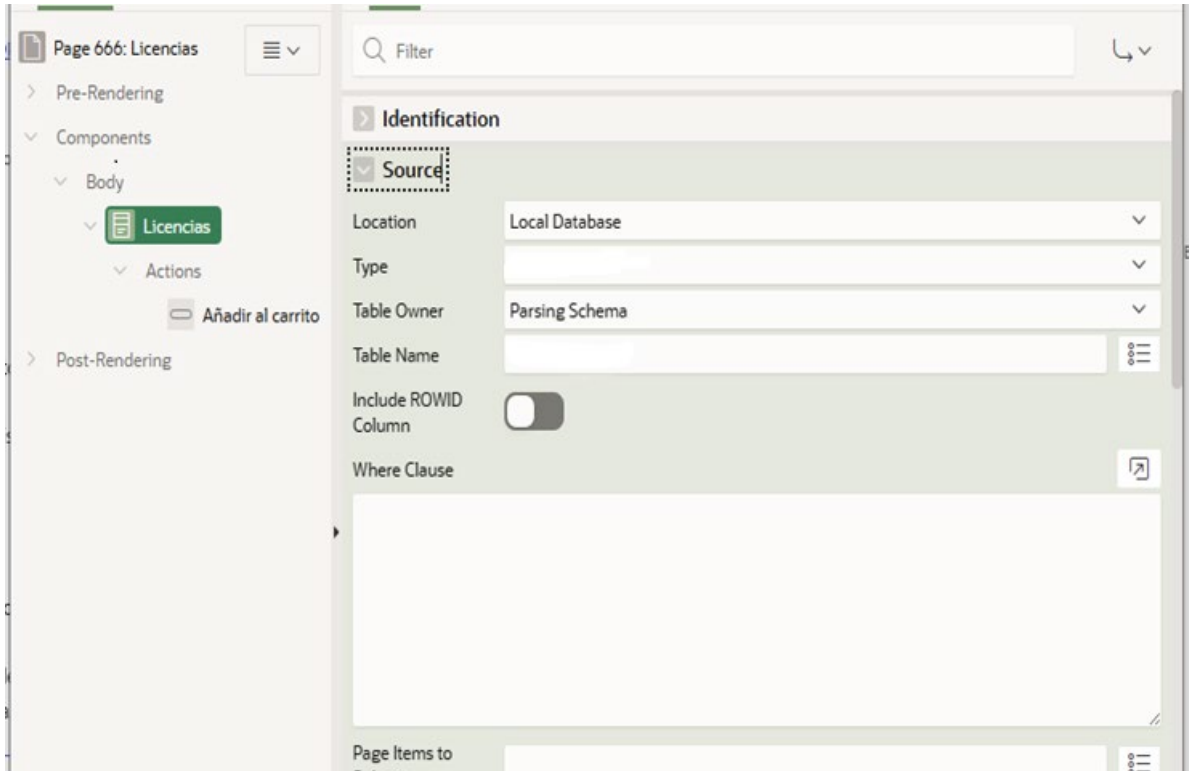
L'aspecte que ha de presentar la pantalla és el següent:

The screenshot shows a web interface for purchasing software licenses. The header is blue with the text 'Compra de licencias sw' and a user profile 'nobody'. The main content area is a grid of seven license cards. Each card has a logo, the license name, a stock quantity, the price, and a blue button labeled 'Añadir al carrito'.

Logo	Nombre	Stock	Precio
W	LICENCIA WORD 2025	450	299.9 euros
X	LICENCIA EXCEL 2020	100	123 euros
A	LICENCIA ADOBE 2022	200	450 euros
W	LICENCIA WORD 2023	25	100 euros
W	LICENCIA WORD 2020	10	25 euros
X	LICENCIA EXCEL 2025	125	300 euros
A	LICENCIA ADOBE 2025	350	790 euros



Resposta 6a: Empleneu tots els camps de l'apartat SOURCE del component CARDS que considereu necessaris per a mostrar les llicències de les categories 1, 2 i 3.





Resposta 6b: Empleneu tots els camps dels apartats CARD, TITLE, BODI i ICON AND BADGE que considereu necessaris perquè l'aspecte de la pantalla siga el que es mostra en la imatge (no cal emplenar els camps CSS Classes)

The image shows the configuration interface for a table component in SAP Fiori. The left sidebar displays the component tree, with 'Licencias' selected under 'Body'. The main configuration area is divided into several sections:

- Appearance:** Contains sub-sections for Card, Title, Subtitle, and Body.
- Card:** Includes fields for CSS Classes, Primary Key Column 1, and Primary Key Column 2.
- Title:** Includes a toggle for Advanced Formatting, a Column dropdown, and a CSS Classes field.
- Subtitle:** (Currently collapsed).
- Body:** Includes a toggle for Advanced Formatting and an HTML Expression field.
- Secondary Body:** (Currently collapsed).
- Icon and Badge:** Expanded section with the following settings:
 - Icon Source: Image BLOB Column
 - Image Column: (empty)
 - Icon CSS Classes: (empty)
 - Icon Position: Start
 - Icon Description: (empty)
 - Badge Column: (empty)
 - Badge Label: Stock
 - Badge CSS Classes: (empty)



Pregunta 7 (3 punts): Es té el *backoffice* de l'aplicació implementat en APEX i es té un INTERACTIVE GRID en què es pot fer el manteniment de les llicències tal com s'observa en la imatge:

Compra de licencias sw nobody

Search: All Text Columns

<input type="checkbox"/>		Id	Nombre	Precio	Categoría	Stock	
<input checked="" type="checkbox"/>		1	LICENCIA WORD 2025	299.9	WORD	450	Subir imagen
<input type="checkbox"/>		4	LICENCIA WORD 2023	100	WORD	25	Subir imagen
<input type="checkbox"/>		5	LICENCIA WORD 2020	25	EXCEL	10	Subir imagen
<input type="checkbox"/>		2	LICENCIA EXCEL 2020	123	ADOBE	100	Subir imagen
<input type="checkbox"/>		6	LICENCIA EXCEL 2025	300	EXCEL	125	Subir imagen
<input type="checkbox"/>		3	LICENCIA ADOBE 2022	450	ADOBE	200	Subir imagen
<input type="checkbox"/>		7	LICENCIA ADOBE 2025	790	ADOBE	350	Subir imagen

1 rows selected Total 7



Resposta 7: Empleneu les propietats del camp CATEGORIA que calguen perquè recupere els valors de la taula i aparega el desplegable amb la descripció de les categories:

The screenshot shows the Oracle APEX configuration interface for a field named 'CATEGORIA'. On the left, a tree view shows the component hierarchy: 'Page 777: Mantenimiento de Llcencia' > 'Body' > 'Mantenimiento de Llcencias' > 'Columns' > 'CATEGORIA'. The 'CATEGORIA' column is highlighted in green. The main configuration panel on the right is divided into several sections:

- List of Values:** Type is set to 'SQL Query'. The 'SQL Query' field is empty. There are toggle switches for 'Display Extra Values' and 'Display Null Value', both currently turned off.
- Cascading List of Values:** The 'Parent Column(s)' field is empty.
- Link:** The 'Target' is set to 'No Link Defined'.
- Source:** Type is set to 'Database Column'. The 'Database Column' and 'Data Type' fields are empty. There are toggle switches for 'Query Only' and 'Primary Key', both currently turned off.
- Master Detail:** The 'Master Column' is set to '- Select -'.
- Default:** The 'Type' is set to '- Select -'.