

 	EXAMEN (OPCIÓN B) REC. SEGUNDA EVALUACIÓN (2020/21)	DWESE – 2.º DAW	
		11-mar-2021	Pág. 1 de 2

Apellidos y nombre: _____

Escribir en **Yii 2 framework** una tienda online de zapatos.

En todo momento, el usuario podrá consultar el catálogo de zapatos que hay en la tienda. Si hay alguno que le interese, puede añadirlo a su carrito. En todo momento podrá consultar su carrito a través de una opción en el menú principal de la aplicación (en la barra superior). También podrá eliminar artículos del carrito o vaciarlo completamente. Si finalmente decide comprar lo que tiene en el carrito, podrá hacerlo pulsando un botón.

Se pide:

1. (1 punto) Crear la base de datos con las siguientes tablas:

a) usuarios (id, nombre, password)

El nombre identifica al usuario y debe ser único.

La contraseña debe estar cifrada mediante el algoritmo de cifrado estándar de Yii 2.

b) zapatos (id, codigo, denominacion, precio)

El código es el código de barras (13 dígitos) y debe ser único.

El precio está en Euros.

c) carritos (id, usuario_id, zapato_id, cantidad)

La cantidad no puede ser negativa.

d) facturas (id, usuario_id, created_at)

La columna created_at contiene el instante de creación de la factura.

e) lineas (id, factura_id, zapato_id, cantidad)

La cantidad no puede ser negativa.

2. (1,5 puntos) Crear los modelos correspondientes y todas las relaciones adecuadas entre ellos, implementando las restricciones indicadas en el punto anterior.

3. (1 punto) Si el usuario está logueado, en el menú principal debe aparecer una opción «Ver carrito (*n*)» siendo *n* la cantidad total de artículos que hay en el carrito del usuario. Si el usuario no está logueado, deberá aparecer «Sin carrito». Al pulsar esa opción, se deberá ir a carritos/ver.

4. (0,5 puntos) Todas las acciones del controlador carritos deben ser accesibles únicamente si el usuario está logueado. En caso contrario, se le redirigirá primero a site/login para que se loguee previamente.

5. (2 puntos) Cuando se entre en la aplicación, la página principal será zapatos/index, donde aparecerá la lista de zapatos que hay en la tienda. Al lado de cada zapato debe aparecer un

botón «Añadir al carrito» que, si se pulsa, añadirá ese zapato al carrito del usuario a través de la acción `carritos/meter/id` (siendo *id* el ID del zapato) y volverá de nuevo a `zapatos/index`. Si ese zapato ya estaba en el carrito, se incrementará en uno su cantidad en el carrito.

6. (2,5 puntos) En la vista `carritos/ver`, mostrar el contenido del carrito. En cada línea de detalle del carrito, se debe indicar el zapato, el precio, la cantidad y el importe ($\text{cantidad} \times \text{precio}$) de esa línea. Al lado aparecerán dos botones «+» y «-» para incrementar y decrementar la cantidad de esa línea (si es mediante AJAX, mejor). Al final debe aparecer el importe total como la suma de los importes de cada línea. Finalmente habrá un botón «Vaciar carrito» que, al pulsarlo, eliminará todos los artículos del carrito a través de la acción `carritos/vaciar`.
7. (1,5 puntos) En la vista `carritos/ver` habrá un botón «Realizar pedido» que, al pulsarlo, creará la factura correspondiente, creando las filas que sean necesarias en las tablas `facturas` y `lineas` y vaciará luego el carrito.

La evaluación de cada ejercicio se llevará a cabo siguiendo la siguiente rúbrica:

% del total	Evaluación
0 %	No entregado, apenas empezado o manifiestamente incorrecto.
25 %	Se esboza un mínimo planteamiento pero no se acerca a una solución funcional.
50 %	Hay una solución incompleta que no cumple todos los requisitos.
75 %	La solución es completa pero ineficiente, poco elegante o mal documentada.
100 %	Solución completa, eficiente, elegante y bien documentada.

Se tendrá en cuenta, especialmente:

- Aplicación coherente del patrón MVC.
- Uso de comentarios adecuados en el código.
- Documentación del mismo en formato PHPDoc.
- Aplicación de los estándares de codificación usados por el framework.
- Elección adecuada de las rutas de acceso a la funcionalidad de la aplicación.
- Formato apropiado de los datos.
- Uso de mecanismos de comunicación asíncrona (AJAX).