

## PRÁCTICA 2 DAI: PHP, MySQL y Laravel

### Desarrollo de la Interfaz Básica. (5 puntos)

En esta práctica se pretende desarrollar una aplicación web que ayude en el desarrollo de la actividad diaria de un **Taller de Reparación de Embarcaciones**. Para gestionar el taller se dispondrá de una base de datos que soporte el trámite de los clientes, embarcaciones, repuestos y facturas. La base de datos será desarrollada en **MySQL** y contará con las siguientes tablas:

- ✚ **Clientes:** datos personales de los clientes.
- ✚ **Embarcaciones:** datos de las embarcaciones que son reparadas en el taller.
- ✚ **Repuestos:** datos de las piezas, los repuestos de las embarcaciones.
- ✚ **Factura:** información general de la factura asociada a una reparación.
- ✚ **Detalle\_Factura:** detalle de las líneas de las facturas.

El nombre de la base de datos será **OBLIGATORIAMENTE:** *taller\_embarcaciones*. Las tablas de la base de datos **MySQL** deberán presentar **EXACTAMENTE** los siguientes campos:

- ✚ **Clientes:** Id\_Cliente, DNI, Nombre, Apellido1, Apellido2, Dirección, C.P., Población, Provincia, Teléfono, E-mail, Fotografía.
- ✚ **Embarcaciones:** Matrícula, Longitud, Potencia (CV), Motor [Diesel, Eléctrico, Gas, Gasolina, Híbrido], Año, Color, Material [Acero, Aluminio, Carbono, Fibra de Vidrio, Polietileno], Id\_Cliente.
- ✚ **Repuestos:** Referencia, Descripción, Importe, Ganancia (Porcentaje de beneficio), Fotografía.
- ✚ **Factura:** Numero\_Factura, Matricula, Mano de Obra (Horas), Precio\_Hora, Fecha\_Emisión, Fecha\_Pago, Base\_Imponible, IVA, Total
- ✚ **Detalle\_Factura:** Id\_Det\_Factura, Numero\_Factura, Referencia, Unidades

### Especificaciones de la Aplicación:

Para la parte estática de la aplicación web se empleará HTML y CSS, mientras que para la parte dinámica se empleará el lenguaje PHP sobre una base de datos MySQL.

La aplicación permitirá **loguearse** mediante **usuario y contraseña**. NO se debe permitir el acceso a ninguna de las funcionalidades de la aplicación sin identificarse previamente. Tras el login, el usuario de podrá realizar las siguientes operaciones:

#### Gestión de Clientes:

Esta opción mostrará un **listado de todos los clientes** de la BD.

Al hacer clic sobre uno de los clientes de la lista, se accederá a la ficha **Detalles del Cliente**, donde podrán modificarse todos los datos de este cliente, excepto el Id del Cliente que será autonumérico.

Además, se mostrará un botón que permita **Añadir un nuevo cliente**.

También se incluirá en este apartado un botón que permita **eliminar clientes**. Se debe poder eliminar un cliente desde la ficha Detalles del Cliente, o bien la aplicación debe permitirnos seleccionar varios clientes de una lista para eliminarlos en lote.

### Gestión de Embarcaciones

Esta opción mostrará un **listado de todas las embarcaciones** de la BD.

Al hacer clic sobre una de las embarcaciones de la lista, se accederá a la ficha **Detalles de la Embarcación**, donde podrán modificarse todos los datos de este barco.

Además, se mostrará un botón que permita **Añadir una nueva Embarcación**.

Para indicar el dueño del barco, la aplicación nos deber permitir seleccionar uno de los clientes de la base de datos. No se podrá introducir una embarcación en la tabla Embarcaciones sin que su dueño esté previamente dado de alta en la tabla Clientes.

También se incluirá en este apartado un botón que permita **eliminar embarcaciones**. Se debe poder eliminar un barco desde la ficha Detalles de la Embarcación, o bien la aplicación debe permitirnos seleccionar varios barcos de una lista para eliminarlos en lote.

### Gestión de Repuestos

Esta opción mostrará un **listado de todos los repuestos** de la BD.

Al hacer clic sobre uno de los repuestos de la lista, se accederá a la ficha **Detalles del Repuesto**, donde podrán modificarse todos los datos de este repuesto.

Además, se mostrará un botón que permita **Añadir un nuevo repuesto**.

También se incluirá en este apartado un botón que permita **eliminar repuestos**. Se debe poder eliminar un repuesto desde la ficha Detalles del Repuesto, o bien la aplicación debe permitirnos seleccionar varios repuestos de una lista para eliminarlos en lote.

Para cada repuesto podrá especificar un **Importe** (en Euros) y un porcentaje de **Ganancia** que es el beneficio sobre el importe que obtendrá el taller al incluirlo en una reparación.

### Gestión de Facturas

Esta opción mostrará un **listado de todas las facturas** de la BD.

Al hacer clic sobre una de las facturas de la lista, se accederá a la ficha **Detalles de la Factura**, donde podrán modificarse todos los datos relativos a esta factura.

Además, se mostrará un botón que permita **Añadir una nueva factura**.

Para indicar la embarcación a la que se le asigna la factura, la aplicación nos deber permitir seleccionar una de las embarcaciones de la base de datos. No se podrá introducir una matrícula en la tabla facturas sin que esta matrícula esté previamente dada de alta en la tabla de Embarcaciones.

También se incluirá en este apartado un botón que permita **eliminar facturas**. Se debe poder eliminar una factura desde la ficha Detalles de la Factura, o bien la aplicación debe permitirnos seleccionar varias facturas de una lista para eliminarlas en lote.

Para introducir las líneas de detalle de la factura, la aplicación web deber permitirnos introducir una o más líneas de factura, indicando para ello la **Referencia** de un repuesto y el número de **Unidades** del repuesto que se incluye en la línea de factura. No se debe poder indicar una referencia de un repuesto que no esté previamente dado de alta en la base de datos.

La **Base Imponible** de la factura se calculará de la siguiente forma:

- ✚ Para cada línea de detalle de factura se sumará el **Importe** más el **porcentaje** de ganancia.
- ✚ A la suma de todas las líneas de detalle se le sumará el montante total de la **Mano de Obra**, teniendo en cuenta el número de horas de trabajo de la factura más el precio de hora que se estime oportuno.

A partir de la base imponible se calculará el IVA (al tipo actual) y el Total de la factura.

### Seguridad de la Aplicación

En cuanto a la **Seguridad** de la aplicación Web. La validación del usuario ante la aplicación debe controlarse mediante **Sesiones**. Cuando el usuario termine sus gestiones en la aplicación debe poder hacer clic en el botón **Cerrar Sesión** (siempre visible en la aplicación) para que la sesión sea destruida. No se debe poder acceder a ninguno de los menús de la aplicación directamente a través de su URL sin hacer previamente login en la aplicación web.

## Desarrollo de la Interfaz de Consultas (2 puntos)

El usuario de la aplicación debe poder realizar las siguientes consultas a la base de datos:

Obtener un **Listado de Clientes** con los siguientes parámetros de búsqueda (el usuario podrá introducir uno, varios o ningún parámetro de búsqueda y la aplicación siempre debe devolver un listado con los clientes que cumplan estos parámetros):

- ✚ Nombre
- ✚ Apellido 1
- ✚ Apellido 2
- ✚ Población
- ✚ Provincia
- ✚ Teléfono

Además, al hacer clic en uno de los clientes del listado se debe acceder a la **Ficha Detalles del Cliente**.

Obtener los siguientes **Listados de Facturas**:

- ✚ Listado de Facturas Emitidas entre dos fechas.
- ✚ Listado de Facturas Pagadas entre dos fechas.
- ✚ Listado de Facturas pendientes de Pago.
- ✚ Listado de Facturas de un cliente determinado. La aplicación deberá permitirnos seleccionar un cliente de la BD y nos mostrará un listado de todas sus facturas.

Además, al hacer clic en una de las facturas del listado se debe acceder a la **Ficha Detalles de la Factura**.

## MVC con Laravel (3 puntos)

Para finalizar, se deberá implementar dos funcionalidades empleando para ello el **framework PHP de código abierto Laravel**. Se aplicará obligatoriamente el patrón Modelo Vista Controlado de manera que:

Al solicitar por el navegador la url [http://localhost/P2\\_Laravel/public/datos\\_embarcacion/XXXX](http://localhost/P2_Laravel/public/datos_embarcacion/XXXX) siendo **XXXX** una cadena de caracteres, la aplicación web debe **mostrar por pantalla todos los datos** de la embarcación cuya Matricula es igual a dicha cadena.

Al solicitar por el navegador la url [http://localhost/P2\\_Laravel/public/actualizar\\_repuesto/X/Y](http://localhost/P2_Laravel/public/actualizar_repuesto/X/Y) siendo **X** e **Y** dos números, la aplicación web debe **actualizar el importe el repuesto** con referencia X, asignándole un importe Y.

Para el desarrollo de estas funcionalidades se empleará **OBLIGATORIAMENTE** el **sistema de rutas, vistas y controladores** de este framework. Para la interacción con la base de datos se deberá emplear el ORM Eloquent.

### Normas de Entrega

---

La entrega de la práctica se hará en un único fichero zip, que incluirá todas las páginas implementadas, ficheros PHP, imágenes y memoria descriptiva de la aplicación desarrollada. La entrega se realizará mediante el apartado Tareas del campus virtual de la asignatura.

**ESTA PRÁCTICA SUPONE EL 16% DE LA NOTA FINAL DE LA ASIGNATURA.**