

## **Trabajo Desarrollo Web Entorno Servidor – 2ª Evaluación**

El objetivo de este trabajo es desarrollar un sistema básico para un restaurante que permita gestionar platos, pedidos y clientes, con funcionalidades para realizar compras falsas a través de distintas empresas de delivery. Se realizarán los modelos, controladores y vistas necesarias, así como se definirán las rutas de la aplicación según lo visto en clase.

### **Estructura de la aplicación:**

El proyecto en VSCode se creará según el patrón de Laravel y se llamará **'TrabajoRestaurante'**:

Archivos obligatorios (se pueden crear todas las carpetas y archivos extra que necesitéis):

- **Modelos** (atributos):
  - Cliente.php (id, nombre, teléfono, tarjetaCredito)
  - EmpresaDelivery.php (id, nombre, email)
  - Plato.php (id, nombre, precio)
  - Pedido.php (id, idCliente, precioTotal)
  - Plato\_Pedido.php (idPlato, idPedido)
- **Vistas:**
  - Inicio.blade.php
  - Platos.blade.php
  - Empresas.blade.php
  - Pedidos.blade.php
  - FormularioCliente.blade.php
  - FinPedido.blade.php
- **Controllers:** asociados a todos los modelos salvo a Plato\_Pedido.
- **Migraciones:** asociadas a todos los modelos, usa el tipado más conveniente para cada atributo, por ejemplo, los distintos ids serán \$table->id().
- **Factories** (referenciados desde **DatabaseSeeder**):
  - ClienteFactory: 10 clientes con datos aleatorios, los números de tarjeta y de teléfono deben seguir su formato característico.
  - EmpresaDeliveryFactory: 10 empresas aleatorias, el email debe ser corporativo.
  - PlatoFactory: 20 platos aleatorios, no es necesario que sean nombres de comida.
- **Requests:**
  - DatosCliente.php
- **Routes:**
  - web.php: las rutas son ilimitadas, cread las que necesitéis.

## Funcionalidad de la aplicación

El trabajo no consiste en hacer una plataforma de compra real y funcional, sino en crear un proyecto en Laravel correcto y básico. Lo importante es que cada vista realice una serie de tareas. A continuación se desarrollarán las tareas que se valorarán:

- **Vista Inicio:**
  - Contendrá el nombre del restaurante, una imagen o varias del mismo y un botón/enlace de acceso hacia Platos.
- **Vista Platos:**
  - Contendrá un header para poder moverte a Platos, Empresas y Pedidos.
  - Contendrá una tabla paginada de todos los platos (5 por página) de la base de datos.
  - Se podrá pulsar en cada plato o en un botón a su lado para añadir el plato al pedido del cliente.
  - Se podrá pulsar en un botón/enlace para moverte a la vista FormularioCliente.
- **Vista FormularioCliente:**
  - Contendrá un formulario para escribir nombre, teléfono y tarjeta de crédito del cliente.
  - Se podrá escoger la empresa de Delivery a utilizar en un desplegable con los nombres de todas las empresas en base de datos.
  - Se validará que el teléfono tiene un formato correcto (9 dígitos, empezando con 6, 7, 8 o 9) y lo mismo para la tarjeta (entre 13 y 19 dígitos).
  - Con los datos introducidos, podrás ir a FinPedido.
- **Vista FinPedido:**
  - Contendrá un mensaje de final de pedido y 2 botones para ir a Inicio o a Platos.
- **Vista Empresas:**
  - Contendrá un header para poder moverte a Platos, Empresas y Pedidos.
  - Contendrá una tabla paginada con todas las empresas recogidas (5 por página) de la base de datos.
- **Vista Pedidos:**
  - Contendrá un header para poder moverte a Platos, Empresas y Pedidos.
  - Contendrá una tabla de los últimos 10 pedidos, mostrando su usuario y el precioTotal del mismo.
- **Bootstrap:**
  - Añadir 5 elementos diferentes que usen estilos de Bootstrap en la página web, por ejemplo, botones, cards, navigationBars o alerts.

## Detalles a tener en cuenta

- Cuando un cliente termine su pedido, en base de datos se deberá guardar un nuevo cliente con los datos introducidos, así como un Pedido y los cambios en Plato\_Pedido necesarios.
  - Ejemplo: el cliente Juan hace un pedido de 1 agua y 1 bocadillo. En BBDD se añadirá el cliente (id, Juan, teléfono, tarjetaCrédito), el pedido (id, precio del agua + el bocadillo) y dos plato\_pedido (idAgua, idPedido)(idBocadillo, idPedido). Si después Juan realizara otro pedido, se añadiría otro cliente nuevo.
- Podéis utilizar todo lo aprendido de PHP del curso, no solamente de la parte de Laravel.
- Tenéis libertad a la hora de pensar las rutas, controladores y métodos de los mismos siempre que se cumpla la funcionalidad de la aplicación.

## Evaluación:

Este trabajo cuenta el 30% de la nota final de la 2ª Evaluación. A la hora de evaluar se tendrán en cuenta los siguientes puntos:

- Correcta y funcional estructura de vistas: +1,5
- Correcta visualización de las tablas de platos, empresas y pedidos: +1
- El proyecto añade correctamente los datos de clientes y pedidos a la base de datos: +2
- Formulario correcto y con las validaciones de datos de cliente y tarjeta añadidas: +2
- Migraciones y factories correctas: +1,5
- Se han añadido los 5 elementos de Bootstrap: +1
- Estética de la aplicación (uso de estilos CSS y HTML bien estructurado): +1

Cualquier indicio fundado de copia hará que el trabajo pase a estar suspendido con un 0. El trabajo es individual, aunque podéis ayudaros entre compañeros en las horas de clase de la asignatura.

## Entrega:

Entregar el trabajo en Aeducar antes del **17 de febrero a las 23:59**.

Se entregará todo el proyecto comprimido en un archivo llamado: **trabajo2ev\_NombreApellido.zip**