

# Documentación TPV FastFood

Especificaciones Técnicas, Instalación Local y Despliegue en Producción

## ◆ 1. Tecnologías de Desarrollo

El sistema ha sido desarrollado utilizando un stack tecnológico moderno, robusto y escalable:

### Laravel 10

Framework PHP (Backend)

### MySQL / MariaDB

Motor de Base de Datos

### Blade & Livewire

Motores de Plantilla y  
Componentes

### Bootstrap 4/5 & CSS3

Diseño Frontend  
Responsivo

## ◆ 2. Funcionalidades del Sistema

El sistema cuenta con módulos integrales para la gestión completa de un restaurante o negocio de comida rápida:

- **Punto de Venta (POS):** Interfaz ágil para la toma de pedidos, facturación e impresión de tickets.
- **Módulo de Cocina (KDS):** Pantalla en tiempo real para que los cocineros visualicen y marquen pedidos como completados.
- **Gestión de Mesas y Zonas:** Control visual de las mesas ocupadas, libres y en proceso de pago.
- **Inventario y Productos:** Control de stock, categorías, y registro de platillos con imágenes y precios.
- **Control de Usuarios y Roles:** Permisos específicos para Administradores, Cajeros, Meseros y Cocineros.

- **Reportes y Estadísticas:** Panel de control (Dashboard) con gráficos de ventas diarias, mensuales y productos más vendidos.
- **Configuración del Negocio:** Personalización del logo, nombre de la empresa, moneda, y porcentaje de impuestos.
- **Mantenimiento (Backups):** Generación y restauración de copias de seguridad de la base de datos directamente desde el panel.

## ✦ 3. Instalación Local en Computadora Nueva

---

Sigue estos pasos para poner a funcionar el sistema en un entorno local (Windows):

### Programas Previos Necesarios

1. Descargar e instalar **XAMPP** (incluye PHP y MySQL).
2. Descargar e instalar **Composer** (Gestor de dependencias de PHP).
3. Un editor de código como **Visual Studio Code**.

### Paso a Paso de Instalación

1. Copia la carpeta del proyecto `crm-tpv-fastfood` dentro del directorio de tu servidor local:  
`C:\xampp\htdocs\`.
2. Abre el panel de control de XAMPP e inicia los módulos de **Apache** y **MySQL**.
3. Abre tu navegador y entra a `http://localhost/phpmyadmin`. Crea una nueva base de datos llamada `tpv_fastfood`.
4. En la carpeta del proyecto, duplica el archivo `.env.example` y renómbralo como `.env`.
5. Abre el archivo `.env` y configura la conexión a la base de datos:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=tpv_fastfood
DB_USERNAME=root
DB_PASSWORD=
```

6. Abre una terminal dentro de la carpeta del proyecto e instala las dependencias:

```
composer install
```

7. Genera la clave de la aplicación:

```
php artisan key:generate
```

8. Importa la base de datos. Puedes hacerlo importando el archivo `bk_basededatos.sql` desde phpMyAdmin, o ejecutando migraciones:

```
php artisan migrate --seed
```

9. Crea el enlace simbólico para las imágenes:

```
php artisan storage:link
```

0. Inicia el servidor de prueba de Laravel:

```
php artisan serve
```

1. Accede al sistema desde tu navegador en: `http://localhost:8000` .

## ◆ 4. Hosting Web Recomendado

---

Para aplicaciones desarrolladas en Laravel, se recomienda un hosting que ofrezca panel de control (cPanel o hPanel), acceso SSH, y soporte para PHP 8.1 o superior. Las opciones más recomendadas son:

- **Hostinger (Plan Premium o Business):** *[Recomendado]* Excelente relación calidad-precio, muy veloz, interfaz fácil de usar (hPanel) y perfectamente compatible con Laravel.
- **HostGator o Namecheap:** Buenas alternativas con cPanel tradicional.
- **DigitalOcean (Droplet):** Para usuarios avanzados que desean configurar su propio servidor Ubuntu (VPS) desde cero.

## ◆ 5. Paso a Paso: Subir a Producción (Hostinger / cPanel)

---

**Nota:** Este método es el más seguro en hostings compartidos, ya que protege los archivos centrales del framework manteniéndolos fuera de la carpeta pública de acceso web.

1. **Preparar el Proyecto:** En tu computadora local, abre la terminal en el proyecto y ejecuta:

```
composer install --optimize-autoloader --no-dev
```

2. **Comprimir Archivos:** Comprime todos los archivos y carpetas del proyecto en un archivo `proyecto.zip`.

3. **Crear Base de Datos en Hosting:** Entra al panel de tu hosting, ve a la sección de Bases de Datos MySQL, crea una base de datos, un usuario y asígnale todos los privilegios. Importa tu archivo `bk_basededatos.sql` a través de phpMyAdmin del hosting.

4. **Subir los Archivos:** Ve al Administrador de Archivos de tu hosting. Sube el archivo `proyecto.zip` a la **raíz de tu cuenta** (un nivel arriba de la carpeta `public_html`).

5. **Extraer y Renombrar:** Extrae el archivo ZIP. Te quedará una carpeta, renómbrala a `tpv-app`.

6. **Configurar Carpeta Pública:** Entra a la carpeta `tpv-app`, selecciona **todo el contenido de la carpeta** `public` y muévelo hacia la carpeta `public_html` de tu hosting.

**7. Vincular Rutas (index.php):** Entra a `public_html` , edita el archivo `index.php` y modifica estas dos líneas para que apunten a la carpeta donde dejaste el núcleo:

```
require __DIR__.'/../tpv-app/vendor/autoload.php';

$app = require_once __DIR__.'/../tpv-app/bootstrap/app.php';
```

**8. Configurar .env:** Ve a la carpeta `tpv-app` , edita el archivo `.env` con las credenciales de la base de datos de producción y cambia:

```
APP_ENV=production
APP_DEBUG=false
APP_URL=https://tudominio.com
```

**9. Generar enlace de imágenes:** Si tienes acceso a Terminal (SSH) en tu hosting, entra a la carpeta `tpv-app` y ejecuta `php artisan storage:link` . Si no tienes SSH, puedes crear una ruta en `web.php` que ejecute el comando `Artisan::call('storage:link');` al visitarla una vez.