



DOCUMENTACIÓN TÉCNICA

AutoTaller Pro

Sistema SaaS de Gestión para Talleres Automotrices

Tecnología, funcionalidades, guía de instalación local y
despliegue en producción.

Framework: Laravel 11 + PHP 8.2+

Base de datos: MySQL 8

Versión: 1.0

Fecha: Julio 2026

1 Tecnología utilizada

Stack de desarrollo del sistema

El sistema está desarrollado como una **aplicación web moderna** siguiendo el patrón Modelo-Vista-Controlador (MVC) de Laravel. No requiere compilación de assets (JavaScript/CSS), lo que facilita enormemente su instalación y despliegue.



Arquitectura de la aplicación

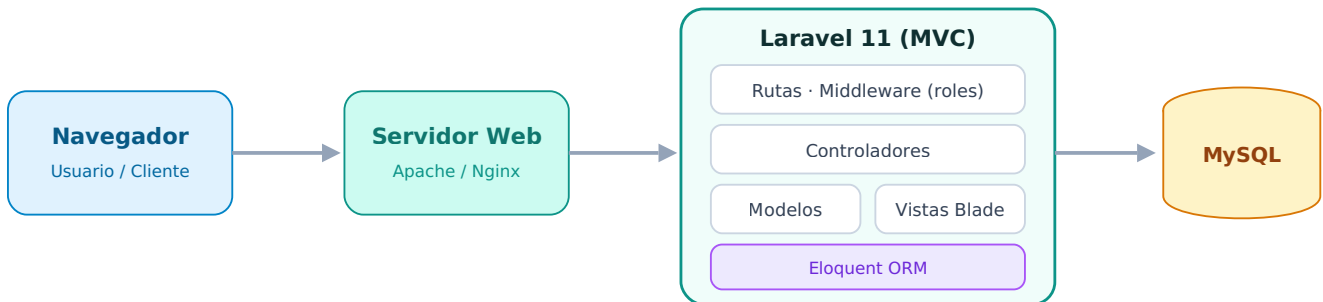


Figura 1 · Flujo de una petición: del navegador al servidor, procesada por Laravel y persistida en MySQL.

DIMENSIÓN DEL PROYECTO	CANTIDAD	DESCRIPCIÓN
Controladores	25	Lógica de cada módulo del sistema
Modelos Eloquent	18	Entidades de negocio mapeadas a tablas
Migraciones	17	Estructura versionada de la base de datos
Vistas Blade	49	Pantallas e interfaces del sistema

2

Funcionalidades del sistema

Módulos y capacidades disponibles

AutoTaller Pro cubre la operación completa de un taller automotriz, más una **capa SaaS** con panel de Super Administrador para gestionar la plataforma como negocio de suscripción.

Operación del taller

- ✓ Órdenes de servicio con servicios y repuestos
- ✓ Cálculo automático de subtotal, IGV y total
- ✓ Cambio de estado y estado de pago
- ✓ Impresión / PDF de la orden
- ✓ Citas / agenda con vista de calendario
- ✓ Catálogo de servicios y mano de obra

Clientes y vehículos

- ✓ Directorio de clientes (persona / empresa)
- ✓ Ficha del cliente con historial y facturación
- ✓ Parque vehicular con datos técnicos
- ✓ Ficha del vehículo con historial de órdenes

Inventario y compras

- ✓ Repuestos con stock y stock mínimo
- ✓ Descuento automático de stock al usarlos
- ✓ Proveedores
- ✓ Compras que reponen inventario
- ✓ Alertas de stock bajo en la campana
- ✓ Valor de inventario en tiempo real

Finanzas

- ✓ Facturación (boletas y facturas) con correlativo
- ✓ Comprobante imprimible / PDF
- ✓ Caja: ingresos, egresos y saldos
- ✓ Registro automático de pagos de órdenes

Administración y seguridad

- ✓ Personal / usuarios con roles
- ✓ Roles y permisos por módulo
- ✓ Perfil de usuario y cambio de contraseña
- ✓ Reportes con filtros y gráficos
- ✓ Exportar a Excel y PDF
- ✓ Importar clientes y repuestos desde CSV
- ✓ Configuración de empresa e IGV
- ✓ Dashboard con KPIs y 4 gráficos

Plataforma SaaS · Super Admin

- ✓ Landing pública de marketing
- ✓ Dashboard SaaS (MRR, ARR, activos)
- ✓ Gestión de talleres suscritos
- ✓ Planes de suscripción
- ✓ Registro de pagos y renovaciones
- ✓ Estados: prueba, activo, suspendido

3

Instalación en una computadora nueva

Paso a paso para Windows

Programas necesarios a descargar e instalar. Con **Laragon** obtienes PHP, MySQL, Apache, Composer y HeidiSQL en un solo instalador, por lo que es la opción recomendada.

PROGRAMA	PARA QUÉ SIRVE	DÓNDE DESCARGAR
Laragon (Full)	PHP 8.2+, MySQL, Apache, Composer y HeidiSQL	laragon.org/download
Composer	Gestor de dependencias de PHP (incluido en Laragon)	getcomposer.org
Navegador web	Ejecutar la aplicación (Chrome, Edge, Firefox)	—
VS Code (opcional)	Editar el código si se requiere	code.visualstudio.com

1 Instalar Laragon

Descarga la versión **Full** desde laragon.org e instálala (siguiente, siguiente, finalizar). Ábrelo y presiona `Start All`.

2 Copiar el proyecto

Copia la carpeta del sistema dentro de `C:\laragon\www\` (por ejemplo `C:\laragon\www\saas_taller-automotriz`).

3 Abrir la terminal

En Laragon, clic en **Terminal** y entra a la carpeta del proyecto con `cd C:\laragon\www\saas_taller-automotriz`.

4 Instalar dependencias y preparar la app

```
# Instala Laravel y sus paquetes
composer install
# Genera la llave de la aplicación
php artisan key:generate
```

5 Crear la base de datos

```
mysql -u root -e "CREATE DATABASE saas_taller_automotriz CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;"
```

O créala visualmente en HeidiSQL (incluido en Laragon) con el nombre `saas_taller_automotriz`.

6 Crear tablas y cargar datos demo

```
php artisan migrate --seed
php artisan db:seed --class=DemoFinanzasSeeder
php artisan db:seed --class=SaasSeeder
```

7

Levantar el servidor

```
php artisan serve
```

Abre `http://localhost:8000` en el navegador.

Credenciales de acceso (demo)

Taller — Admin: **admin@taller.com** · Mecánico: **carlos@taller.com**

Super Admin (SaaS): **superadmin@autotaller.com** · Contraseña para todos: **password**

Consejo

Si algo no refresca tras cambios, ejecuta `php artisan optimize:clear` y recarga con Ctrl+F5.

4

Hosting web recomendado

Dónde publicar tu aplicación

Una app Laravel necesita un hosting con **PHP 8.2+**, **MySQL** y **acceso por consola (SSH)**. A continuación, las mejores opciones según presupuesto y nivel técnico (precios aprox. en USD, julio 2026).

PROVEEDOR	TIPO	IDEAL PARA	PRECIO APROX.	DIFICULTAD
Hostinger VPS	VPS (Ubuntu + plantilla Laravel)	Producción real, buen precio	desde \$5-7/mes	Media
Hostinger Hosting	Compartido	Proyectos pequeños / pruebas	desde \$2.99/mes	Baja
Laravel Cloud / Forge	Administrado	Equipos técnicos, despliegue pro	desde \$12-20/mes	Media-Alta
DigitalOcean	VPS (Droplet)	Escalabilidad y control total	desde \$6/mes	Alta
Railway / Render	PaaS	Despliegue rápido desde Git	gratis / desde \$5/mes	Baja-Media

RECOMENDACIÓN

Hostinger VPS con plantilla Laravel

Mejor relación precio-facilidad para talleres y pymes en LATAM. Incluye una plantilla **Ubuntu 22.04 + Laravel** que instala PHP, servidor web y dependencias automáticamente, soporte en español y precios accesibles. Es la opción con la que continúa la guía.

Nota: el hosting **compartido** de Hostinger también funciona para apps sencillas, pero el **VPS** ofrece mejor rendimiento, acceso SSH completo y menos limitaciones para Laravel.

5

Despliegue en Hostinger

De tu computadora a producción



Figura 2 · Flujo general de despliegue a producción.

Vía rápida (recomendada) — VPS con plantilla

En hPanel: **VPS** → **Administrar** → **SO y Panel** → **Sistema Operativo** → **Aplicación** → **Laravel** → **Cambiar SO**. Esto deja el servidor listo con PHP, Nginx y Composer. Luego solo subes tu código y configuras la BD (pasos 2–6).

1 Contratar el VPS y elegir la plantilla Laravel

Adquiere un plan VPS (KVM 1 o 2 es suficiente) y en el asistente selecciona la plantilla **Ubuntu 22.04 + Laravel**.

2 Subir el proyecto

Sube los archivos por **Git** (recomendado) o con un cliente FTP como **FileZilla** / el Administrador de archivos de hPanel. No subas `vendor/` ni `.env`.

```
# Opción Git (dentro del servidor por SSH)
git clone TU_REPOSITORIO.git
cd saas_taller-automotriz
```

3 Crear la base de datos MySQL

En hPanel crea una base de datos y un usuario con todos los privilegios. Anota **nombre**, **usuario** y **contraseña**.

4 Configurar el archivo .env

```
cp .env.example .env
# Edita con nano y coloca tus datos:
APP_ENV=production
APP_DEBUG=false
APP_URL=https://tudominio.com
DB_DATABASE=nombre_bd
DB_USERNAME=usuario_bd
DB_PASSWORD=contraseña_bd
```

5 Instalar dependencias y migrar

```
composer install --no-dev --optimize-autoloader
php artisan key:generate
php artisan migrate --force --seed
php artisan db:seed --class=SaasSeeder --force
php artisan storage:link
php artisan config:cache
```

6 Apuntar el dominio a la carpeta public

Configura el **Document Root** del dominio hacia la carpeta `/public` del proyecto y activa el **certificado SSL** gratuito (Let's Encrypt) desde hPanel. Abre tu dominio y verifica que cargue.

Hosting compartido (alternativa)

Si usas hosting compartido: sube el proyecto, mueve el contenido de `/public` a `public_html` y ajusta las rutas en `index.php`, crea la BD, configura `.env` y ejecuta `composer install --no-dev` y `php artisan migrate --force` desde el SSH del panel.

Checklist final de producción

✓ APP_DEBUG=false ✓ SSL activo (https) ✓ Document Root en /public ✓ Permisos de escritura en storage/ y bootstrap/cache/ ✓ RespalDOS de BD programados

Fuentes consultadas: Hostinger (VPS Laravel, tutorial de despliegue), WebsitePlanet y Bitcatcha (comparativas de hosting Laravel 2026).