

TRACKING

Manual de Instalación

Versión 1.0 · Agosto 10 de 2026

1. Introducción

Este manual guía la instalación completa de la aplicación Tracking desde cero. Los pasos están ordenados por dependencia: cada sección asume que la anterior se ha completado.

Tracking es una aplicación web PHP compuesta por:

- **Backend PHP** con API REST accesible en `backend/api.php`.
- **Frontend** HTML/CSS/JS sobre la plantilla Gentelella (Bootstrap). Se distribuye ya compilado; no requiere ningún paso de build.
- **Arquitectura de datos multiconexión** descrita a continuación.

1.1 Arquitectura de bases de datos

El sistema trabaja con dos niveles de base de datos claramente diferenciados. Entender esta separación es imprescindible antes de instalar:

Base de datos	Cantidad	Función
trackingweb	Una, obligatoria	Base de control. Almacena usuarios, conexiones, páginas y permisos. Es la única que se importa durante la instalación.
tracking_*	N, opcionales	Bases de datos de explotación (pesadas, partidas, bodegas). Ya existen en el sistema del cliente.
olibest	Opcional	Misma estructura que una base tracking, con datos propios. Se registra como una conexión más.

La aplicación no conoce las bases de explotación por configuración: las lee de la tabla `conexion` de `trackingweb`. Dar de alta una nueva base de datos se hace desde la propia aplicación, no editando archivos. Véase la sección 10.2.

i Para el detalle de las decisiones de arquitectura, consulte el **Manual Técnico (ADR) (EN PROCESO)**. Para el uso diario de los módulos, consulte la **Guía de Uso (EN PROCESO)**. Este manual cubre exclusivamente la puesta en marcha.

2. Requisitos

Herramienta	Versión	Notas
PHP	8.1 o superior	Extensiones requeridas: mysqli, mbstring, openssl, json
MariaDB	10.5	Requisito estricto: véase el aviso inferior
Composer	2.x	Gestor de dependencias PHP. Incluido en XAMPP >= 8.1
Servidor web	Apache 2.4 / Nginx	Apache incluido en XAMPP
Sistema operativo	Windows / Linux / macOS	Sin restricción

⚠ **MariaDB es obligatorio; MySQL no es válido.** Se utilizan funciones nativas de MariaDB en ciertos procedimientos

ℹ Si ya dispone de un entorno PHP 8.1 + MariaDB 10.5 (servidor propio, VPS, etc.), puede saltar directamente al paso 3.

3. Obtener el proyecto

Copie la carpeta **tracking** en el directorio raíz de su servidor web.

Con XAMPP (Windows)

```
C:\xampp\htdocs\tracking\
```

Con otro servidor local (Linux)

```
/var/www/html/tracking/    ← Apache
/srv/http/tracking/        ← Nginx
```

En un servidor externo (VPS / hosting)

Suba la carpeta al directorio público del dominio (normalmente **public_html/** o **www/**) mediante FTP/SFTP o el gestor de archivos del panel de control.

ℹ Todas las rutas de este manual usan XAMPP en Windows como referencia. Adapte las rutas si utiliza otro entorno.

4. Dependencias PHP (Composer)

El proyecto usa Composer para gestionar librerías externas (**vlucas/phpdotenv**, **guzzlehttp/guzzle**, **phpunit/phpunit**). La carpeta **vendor/** ya viene incluida en el proyecto, por lo que en condiciones normales no necesita hacer nada.

Solo tendrá que ejecutar Composer si la carpeta **vendor/** está vacía o ha sido eliminada:

1. Abra una terminal (CMD o PowerShell en Windows) en la raíz del proyecto.

```
cd C:\xampp\htdocs\tracking
```

2. Ejecute la instalación de dependencias.

```
composer install
```

i Si no tiene Composer instalado, descárguelo desde <https://getcomposer.org/download/> e instálelo. Una vez instalado, el comando estará disponible globalmente en la terminal.

5. Configuración del archivo .env

El archivo `backend/.env` contiene las credenciales y la configuración sensible del sistema. Ya existe con la estructura correcta: ábralo con cualquier editor de texto y ajuste los valores.

⚠ No cree un archivo nuevo. Edite el que ya existe en `backend/.env`. Si no existiera, use `backend/.env.example` como plantilla.

5.1 Campos del archivo

Campo	Descripción	Ejemplo
DB_HOST	Dirección del servidor de base de datos	localhost
DB_NAME_NEW	Nombre de la base de control	trackingweb
DB_USER	Usuario de MariaDB	root
DB_PASS	Contraseña de MariaDB	root
MASTER_KEY	Clave maestra de cifrado (ver 5.2)	(generada)
SUPER_USER	Login del superusuario de emergencia	__tracking_master__
SUPER_SECRET	Hash de acceso del superusuario (ver 5.3)	(generado)
SUPER_SECRET_RAW	Token en texto plano (ver 5.3)	(generado)
SUPER_SECRET_HASH	Hash del token (ver 5.3)	(generado)
SUPER_TOKEN_TTL	Tiempo de vida del token, en minutos	15

i Las credenciales de las bases de explotación **ya no se configuran aquí**. Desde la versión multiconexión no existe el campo `DB_NAME`: las bases tracking se registran desde la aplicación (sección 10.2).

5.2 Clave maestra de cifrado (MASTER_KEY)

La **MASTER_KEY** es la clave con la que el sistema cifra las credenciales de las conexiones almacenadas en la tabla `conexion`. El archivo se distribuye con una clave por defecto, sustituir significa tener que reinsertar todas las conexiones (descrito en el punto 10.2). Para generar una nueva clave ejecute:

```
cd C:/xampp/htdocs/tracking
php backend/utils/generar_master_key.php
```

Dicho valor insertarlo en `MASTER_KEY` del `.env`

⊖ No modifique la MASTER_KEY una vez haya registrado conexiones. Las credenciales cifradas con la clave anterior dejarían de poder descifrarse y todas las conexiones dejarían de funcionar. Si necesita rotarla, primero anote las credenciales, cámbiela y vuelva a registrarlas.

5.3 Generar los valores de Super Acceso

El superusuario es una cuenta de emergencia que vive exclusivamente en el archivo `.env`, sin registro en base de datos. Es el único perfil que puede acceder al sistema sin tener ninguna conexión asignada, por lo que resulta imprescindible en el primer arranque.

Se encuentran unos valores por defecto ya creados para el super usuario:

usuario: `__tracking_master__` **contraseña:** `?Contrasena_muy_segura93`

En caso de querer genera una nueva contraseña los valores necesarios se generan con los scripts incluidos. Ejecute los comandos desde la raíz del proyecto:

Paso 1 — Generar `SUPER_SECRET_RAW` y `SUPER_SECRET_HASH`

```
cd C:\xampp\htdocs\tracking
php backend/utils/setup_env.php
```

El script imprimirá dos valores. Cópielos en el `.env` en sus campos correspondientes:

- El primer valor → campo `SUPER_SECRET_RAW`
- El segundo valor → campo `SUPER_SECRET_HASH`

Paso 2 — Generar `SUPER_SECRET`

Ejecute el siguiente comando sustituyendo `tu-contraseña` por la contraseña que desee asignar al superusuario:

```
php backend/utils/generar_hash_login.php tu-contraseña
```

El script imprimirá un valor: cópielo en el campo `SUPER_SECRET`.

i El campo `SUPER_USER` puede ser cualquier cadena que no coincida con un login real de la tabla `usuario_web`; por ejemplo `__tracking_master__`.

⚠ El archivo `.env` está protegido por `.htaccess` frente a peticiones HTTP externas. Nunca lo suba a un repositorio público.

6. Configuración funcional (config.json)

Además del `.env`, el sistema dispone de un archivo de configuración funcional que gobierna qué opciones se muestran en la interfaz y qué comprobaciones se ejecutan. Se encuentra en:

```
backend/variable/config.json
```

El proyecto incluye `backend/variable/config_template.json` como plantilla de referencia con todas las claves y sus valores por defecto así como opciones existentes aunque no utilizadas aún. Si el archivo `config.json` no existiera, cree una copia de la plantilla con ese nombre.

6.1 Claves disponibles

Clave	Tipo	Función
<code>mostrarOpcionesPesada</code>	enum	Controla qué acciones aparecen en el desplegable de cada pesada: solo Introducir/Eliminar, o todas las opciones.
<code>mostrarOpcionesPartida</code>	enum	Controla qué acciones aparecen en el desplegable de cada partida: solo Introducir/Eliminar, o todas las opciones.

Clave	Tipo	Función
AdminCreaAdmin	bool	Permite que un administrador cree otros administradores.
comprobarAcopladas	bool	Activa la comprobación de pesadas acopladas al filtrar en el listado de partidas.

Cada clave se define con su **tipo**, **label**, **descripcion**, un **default** y el **valor** activo. Para modificar el comportamiento del sistema basta con cambiar el campo **valor**.

i Esta configuración es por instalación: aplica a todo el sistema y a todas las conexiones por igual. No es posible configurar valores distintos por base de datos.

7. Base de datos de control (trackingweb)

Es la única base que debe importar durante la instalación. Contiene el control de acceso completo del sistema.

7.1 Contenido

Tabla	Descripción
usuario_web	Usuarios con acceso a la aplicación web
conexion	Bases de datos tracking registradas y sus credenciales cifradas
paginas	Módulos y páginas disponibles en el sistema
permisos	Tripleta usuario ↔ página ↔ conexión que define el acceso

7.2 Importación

Importe el archivo con HeidiSQL (recomendado con XAMPP) o con cualquier cliente MariaDB:

1. Abra HeidiSQL y conéctese a su servidor MariaDB.
2. Menú superior: Archivo → Ejecutar archivo SQL.
3. Seleccione el archivo `importacion_BD/trackingweb_create.sql`.
4. Pulse Abrir. La base de datos y todas las tablas se crearán automáticamente.

i No hace falta crear la base de datos manualmente antes de importar: el archivo `.sql` ya contiene la instrucción `CREATE DATABASE`.

7.3 Qué incluye la importación

El archivo no crea solo la estructura: deja el sistema listo para el primer acceso con los siguientes datos:

- **Las páginas del sistema** ya pobladas. Si un usuario no ve entradas en el menú lateral, será por falta de permisos, no por falta de páginas.
- **Un administrador de prueba** (`adminPrueba`) con sus permisos ya asignados. Contraseña `admin` con acceso a la conexión `olibest`.
- **Cinco conexiones preregistradas**: `olibest`, `tracking`, `tracking_1`, `tracking_2` y `olibest_ORIG`. Únicamente `olibest` está habilitada.

⚠ Las conexiones preregistradas apuntan a nombres de base de datos de ejemplo. **Debe editarlas desde el panel de conexiones** para que apunten a sus bases reales (sección 10.2). Una conexión mal configurada impedirá el acceso a los módulos de explotación.

8. Bases de datos de explotación (tracking / olibest)

Son las bases ya existentes en su sistema. No requieren importar ninguna estructura: basta con que existan y sean accesibles con las credenciales que registre en el panel de conexiones.

i Para **olibest** no hay ningún paso adicional: comparte estructura con las bases tracking y solo necesita existir para ser registrada como una conexión más.

8.1 Scripts obligatorios en cada base

Este paso es imprescindible y se omite con facilidad. Los módulos de partidas requieren objetos de base de datos que no forman parte de la estructura original. Debe ejecutar los dos scripts siguientes en TODAS y CADA UNA de las bases de explotación que vaya a registrar:

Archivo	Contenido
reajustar_pesadas.sql	Procedimiento almacenado que reajusta las pesadas de una o varias partidas.
OptimizadoPartida.sql	Índices y optimizaciones necesarios para el rendimiento del módulo de partidas.

1. En HeidiSQL, seleccione en el árbol lateral la base de explotación destino (por ejemplo, olibest).
2. Ejecute `importacion_BD/reajustar_pesadas.sql`.
3. Ejecute `importacion_BD/OptimizadoPartida.sql`.
4. Repita el proceso para cada base adicional que registre en el futuro.

⊖ Verifique la base seleccionada antes de ejecutar. Si omite estos scripts en una base, el módulo de partidas funcionará parcialmente y el botón de reajuste de pesadas devolverá un error de SQL.

9. Servidor web

9.1 XAMPP (recomendado en Windows)

1. Descargue e instale XAMPP desde <https://www.apachefriends.org/>
2. Abra el Panel de Control de XAMPP.
3. Pulse Start en el módulo Apache.
4. Pulse Start en el módulo MySQL.
5. Compruebe que la carpeta del proyecto está en `C:\xampp\htdocs\tracking\`

i XAMPP incluye PHP 8.x, Apache, MariaDB y phpMyAdmin. No necesita instalar nada más para un entorno local.

9.2 Otras opciones

Entorno	Sistema	Notas
Laragon	Windows	Alternativa ligera a XAMPP. Incluye PHP, Apache/Nginx y MariaDB
WAMP	Windows	Similar a XAMPP
MAMP	macOS	Equivalente para Mac
Apache + PHP	Linux	Instalar: <code>apache2</code> , <code>php8.1</code> , <code>php8.1-mysqli</code> , <code>php8.1-mbstring</code>

Entorno	Sistema	Notas
Nginx + PHP-FPM	Linux / VPS	Requiere configuración de server block. Recomendado en producción
Hosting compartido	Cualquiera	Subida por FTP. Verifique PHP 8.1 y la extensión mysqli

9.3 Archivos .htaccess

El proyecto incluye archivos .htaccess preconfigurados. No es necesario modificarlos en una instalación estándar. Realizan las siguientes funciones:

- Bloquean el acceso HTTP directo al archivo `.env`.
- Permiten acceder a la aplicación directamente desde `localhost/tracking`, redirigiendo a la pantalla de login.

⚠ Si utiliza Nginx, los archivos `.htaccess` no son compatibles. Deberá traducir las reglas a bloques `location {}`. Asimismo, en Apache asegúrese de que `AllowOverride All` está activo en `httpd.conf`.

10. Primer acceso y puesta en marcha

Con todo lo anterior completado, abra el navegador y acceda a:

```
http://localhost/tracking
```

Debería ver la pantalla de login de la aplicación.

10.1 Credenciales iniciales

Tras importar `trackingweb_create.sql` dispone de los siguientes accesos:

Usuario	Contraseña	Rol	Definido en	Acceso
<code>__tracking_master__</code>	(la definida en 5.3)	Super usuario	<code>.env</code>	Total, sin conexiones
<code>adminPrueba</code>	<code>admin</code>	Administrador	<code>trackingweb</code>	<code>olibest</code>

⚠ **Un usuario o administrador necesita al menos una conexión asignada para poder acceder.** El superusuario es el único perfil que puede entrar sin ninguna conexión, precisamente para poder configurarlas desde cero sin necesidad de utilizar SQL puro.

➖ **Cambie las credenciales de prueba antes de poner el sistema en producción.**

10.2 Registrar las bases de datos (panel de conexiones)

Este es el paso que sustituye a la antigua configuración por archivo. Se realiza desde la propia aplicación:

1. Acceda con el superusuario o con un usuario de rol administrador.
2. Abra el panel de conexiones desde el menú lateral.
3. Edite las conexiones preregistradas para que apunten a sus bases reales, o cree las que necesite.
4. Habilite únicamente las conexiones que vayan a utilizarse.
5. Compruebe que cada base registrada tiene ejecutados los scripts de la sección 8.1.

10.3 Crear usuarios y asignar permisos

Desde el panel de administración, disponible para los roles administrador y superusuario, se gestionan los usuarios y sus permisos. El permiso es una tripleta: define qué usuario accede a qué página sobre qué conexión.

i El detalle del uso de estos paneles se documenta en la **Guía de Uso**. Aquí se indica únicamente su existencia como paso final de la instalación.

10.4 Lista de verificación

	Comprobación
☐	La carpeta tracking está en htdocs/ (o en el raíz web correspondiente)
☐	Apache y MariaDB están arrancados
☐	backend/.env tiene todos los campos rellenos, con MASTER_KEY propia
☐	backend/variable/config.json existe y tiene las claves esperadas
☐	La base trackingweb ha sido importada correctamente
☐	Cada base de explotación tiene ejecutados reajustar_pesadas.sql y OptimizadoPartida.sql
☐	Las conexiones del panel apuntan a bases reales y están habilitadas
☐	La pantalla de login carga sin errores
☐	Un usuario con conexión asignada puede entrar y ver sus módulos

11. Solución de problemas frecuentes

Síntoma	Causa probable	Solución
Pantalla en blanco o error 500	Error de PHP oculto	Revise el log de Apache (xampp/apache/logs/error.log) o active display_errors temporalmente
"No autenticado" en todas las llamadas	La sesión PHP no se inicia	Verifique que session.save_path tiene permisos de escritura
Error de base de datos al hacer login	Credenciales incorrectas en .env	Revise DB_HOST, DB_USER, DB_PASS y DB_NAME_NEW. Pruebe la conexión desde HeidiSQL con los mismos datos
El usuario entra pero no ve módulos	Faltan permisos o conexión asignada	Asigne al usuario al menos una conexión y los permisos de página desde el panel de administración
Una conexión concreta falla	Credenciales cifradas con otra MASTER_KEY	Vuelva a registrar la conexión desde el panel. Véase el aviso de la sección 5.2
El reajuste de pesadas da "Error sql"	Falta el procedimiento almacenado	Ejecute reajustar_pesadas.sql sobre esa base concreta (sección 8.1)
El módulo de partidas va muy lento	Faltan los índices	Ejecute OptimizadoPartida.sql sobre esa base concreta
La página carga pero sin estilos	Ruta de assets incorrecta	Verifique que las carpetas de assets del frontend existen y no están vacías

Síntoma	Causa probable	Solución
Error 403 al acceder	.htaccess bloqueando el acceso	Asegúrese de que AllowOverride All está activo en httpd.conf
vendor/ no encontrado	Composer no ejecutado	Ejecute composer install desde la raíz del proyecto