

Documentation PROJECT SUPPORT

GPS & TRANSPORTATION

INDEX

MODEL BD.....	2
MODELO ENTIDAD RELACIÓN.....	8
DIAGRAMA DE FLUJO.....	11
Flujo de servicio basado en la BD.....	14
12. Flujo resumido en pasos.....	18
Resumen.....	19
Flujo resumido en pasos.....	19
FLUJO EN INSERCIÓN DE BD.....	20
2. Insertar usuarios.....	20
3. Insertar empresas.....	20
RECOMENDACIONES DE USO CON LARAVEL + VUE + PINIA.....	24
Recomendaciones.....	24
Realtime / Actualizaciones en tiempo real.....	25

MODEL BD

-- ROLES Y USUARIOS

```
CREATE TABLE roles (
    id SERIAL PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL
);
```

```
CREATE TABLE usuarios (
    id SERIAL PRIMARY KEY,
    nombre VARCHAR(150) NOT NULL,
    telefono VARCHAR(20),
    email VARCHAR(150) UNIQUE NOT NULL,
    password TEXT NOT NULL,
    activo BOOLEAN DEFAULT true,
    ultimo_login TIMESTAMP,
    rol_id INT REFERENCES roles(id) ON DELETE SET NULL
);
```

-- EMPRESAS Y UNIDADES

```
CREATE TABLE empresas (
    id SERIAL PRIMARY KEY,
    nombre VARCHAR(150) NOT NULL,
    direccion TEXT
);
```

```
CREATE TABLE unidades (
    id SERIAL PRIMARY KEY,
    empresa_id INT REFERENCES empresas(id) ON DELETE CASCADE,
    tipo VARCHAR(100) NOT NULL,
```

caracteristicas TEXT,

```
ignicion_movimiento VARCHAR(50) CHECK (ignicion_movimiento IN ('IGNICION', 'MOVIMIENTO'))  
);
```

-- TABLAS MAESTRAS

```
CREATE TABLE lectoras (  
    id SERIAL PRIMARY KEY,  
    tipo VARCHAR(50) CHECK (tipo IN ('MAGNETICA', 'RFID', 'OTRO'))  
);
```

```
CREATE TABLE discos_duros (  
    id SERIAL PRIMARY KEY,  
    tipo VARCHAR(50) CHECK (tipo IN ('SSD', 'HDD', 'OTRO'))  
);
```

```
CREATE TABLE cables (  
    id SERIAL PRIMARY KEY,  
    estado VARCHAR(50) CHECK (estado IN ('NUEVO', 'USADO', 'DANADO'))  
);
```

```
CREATE TABLE gps_modelos (  
    id SERIAL PRIMARY KEY,  
    modelo VARCHAR(100)  
);
```

-- INVENTARIO EXTERNO

```
CREATE TABLE piezas (
    id SERIAL PRIMARY KEY,
    tipo VARCHAR(50) NOT NULL,
    id_opcion INT,
    nombre VARCHAR(150),
    descripcion TEXT,
    stock INT DEFAULT 0,
        CONSTRAINT piezas_tipo_check CHECK (tipo IN
('LECTORA','DISCO_DURO','CABLE','GPS'))
);
```

-- INVENTARIO INTERNO POR UNIDAD

```
CREATE TABLE inventario_interno (
    id SERIAL PRIMARY KEY,
    unidad_id INT REFERENCES unidades(id) ON DELETE CASCADE,
    pieza_id INT REFERENCES piezas(id) ON DELETE CASCADE,
    cantidad INT DEFAULT 1,
    danado BOOLEAN DEFAULT false,
    fecha_actualizacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

-- REGISTRO HISTÓRICO DE PIEZAS

```
CREATE TABLE unidad_pieza_registro (
    id SERIAL PRIMARY KEY,
    unidad_id INT REFERENCES unidades(id) ON DELETE CASCADE,
    inventario_id INT REFERENCES inventario_interno(id) ON DELETE CASCADE,
    cantbus VARCHAR(50),
    danado BOOLEAN DEFAULT false,
    fecha_actualizacion TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    usuario_id INT REFERENCES usuarios(id),
    motivo_cambio TEXT);
```

-- FALLAS Y LOGS

```
CREATE TABLE fallas (
    id SERIAL PRIMARY KEY,
    descripcion TEXT NOT NULL
);

CREATE TABLE log_fallas (
    id SERIAL PRIMARY KEY,
    unidad_id INT REFERENCES unidades(id),
    falla_id INT REFERENCES fallas(id),
    descripcion TEXT,
    fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    activo BOOLEAN DEFAULT true
);
```

-- SERVICIOS

```
CREATE TABLE servicios (
    id SERIAL PRIMARY KEY,
    usuario_id_levanta INT REFERENCES usuarios(id),
    usuario_id_aprueba INT REFERENCES usuarios(id),
    unidad_id INT REFERENCES unidades(id),
    cambios TEXT,
    items TEXT,
    comentarios TEXT,
    fecha_alta DATE DEFAULT CURRENT_DATE,
    fecha_liberacion DATE,
    estado VARCHAR(50),
    domiciliado BOOLEAN DEFAULT false,
    latitud DECIMAL(10,6),
    longitud DECIMAL(10,6),
    pendiente BOOLEAN DEFAULT false,
    reasignado_admin BOOLEAN DEFAULT false
);
```

```
CREATE TABLE servicio_pendiente (  
    id SERIAL PRIMARY KEY,  
    servicio_id INT REFERENCES servicios(id) ON DELETE CASCADE,  
    usuario_id_admin INT REFERENCES usuarios(id),  
    tecnico_id INT REFERENCES usuarios(id),  
    estado VARCHAR(50),  
    fecha_registro TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

```
CREATE TABLE servicio_falla (  
    servicio_id INT REFERENCES servicios(id) ON DELETE CASCADE,  
    falla_id INT REFERENCES fallas(id) ON DELETE CASCADE,  
    PRIMARY KEY (servicio_id, falla_id)  
);
```

```
CREATE TABLE servicio_pieza (  
    servicio_id INT REFERENCES servicios(id) ON DELETE CASCADE,  
    pieza_id INT REFERENCES piezas(id),  
    cantidad INT DEFAULT 1,  
    PRIMARY KEY (servicio_id, pieza_id)  
);
```

```
CREATE TABLE servicio_historial (  
    id SERIAL PRIMARY KEY,  
    servicio_id INT REFERENCES servicios(id),  
    usuario_id INT REFERENCES usuarios(id),  
    accion TEXT,  
    fecha TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

-- EVIDENCIAS

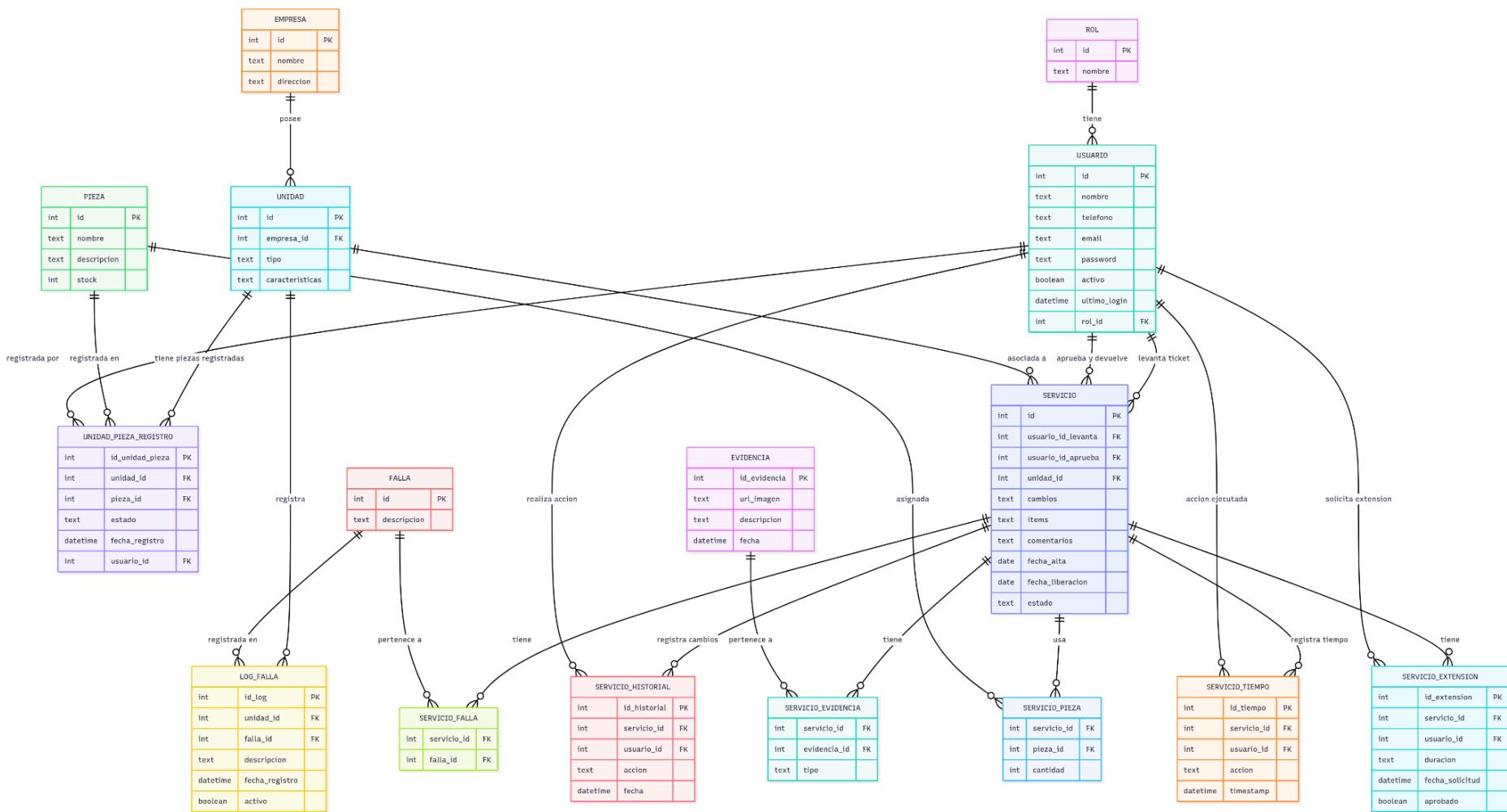
```
CREATE TABLE evidencias (
    id SERIAL PRIMARY KEY,
    url_imagen TEXT,
    descripcion TEXT,
    fecha TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

```
CREATE TABLE servicio_evidencia (
    servicio_id INT REFERENCES servicios(id),
    evidencia_id INT REFERENCES evidencias(id),
    tipo VARCHAR(50),
    PRIMARY KEY (servicio_id, evidencia_id)
);
```

-- TIEMPOS Y EXTENSIONES

```
CREATE TABLE servicio_tiempo (
    id SERIAL PRIMARY KEY,
    servicio_id INT REFERENCES servicios(id),
    usuario_id INT REFERENCES usuarios(id),
    accion TEXT,
    timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

```
CREATE TABLE servicio_extension (
    id SERIAL PRIMARY KEY,
    servicio_id INT REFERENCES servicios(id),
    usuario_id INT REFERENCES usuarios(id),
    duracion TEXT,
    fecha_solicitud TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    aprobado BOOLEAN DEFAULT false
);
```



MODELO ENTIDAD RELACIÓN

erDiagram

```
%% Roles y usuarios
ROL {
    int id PK
    text nombre
}

USUARIO {
    int id PK
    text nombre
    text telefono
    text email
    text password
    boolean activo
    datetime ultimo_login
    int rol_id FK
}

%% Empresas y unidades
EMPRESA {
    int id PK
    text nombre
    text direccion
}

UNIDAD {
    int id PK
    int empresa_id FK
    text tipo
    text caracteristicas
    ENUM ignicion_movimiento
}

%% Tablas maestras compartidas
LECTORAS {
    int id PK
    ENUM tipo
}
```

```
DISCOS_DUROS {
    int id PK
    ENUM tipo
}

CABLES {
    int id PK
    ENUM estado
}

GPS_MODELOS {
    int id PK
    ENUM modelo
}

%% Inventario externo (global)
PIEZA {
    int id PK
    ENUM tipo
    int id_opcion FK
    text nombre
    text descripcion
    int stock
}

%% Inventario interno por unidad
INVENTARIO_INTERNO {
    int id PK
    int unidad_id FK
    int pieza_id FK
    int cantidad
    boolean danado
    datetime fecha_actualizacion
}

%% Registro histórico de piezas por unidad
UNIDAD_PIEZA_REGISTRO {
    int id_unidad_pieza PK
    int unidad_id FK
    int inventario_id FK
    ENUM cantbus
    boolean danado
    datetime fecha_actualizacion
```

```
        int usuario_id FK
        text motivo_cambio
    }

FALLA {
    int id PK
    text descripcion
}

LOG_FALLA {
    int id_log PK
    int unidad_id FK
    int falla_id FK
    text descripcion
    datetime fecha_registro
    boolean activo
}

%% SERVICIO con geolocalización, domiciliado y pendientes
SERVICIO {
    int id PK
    int usuario_id_levanta FK
    int usuario_id_aprueba FK
    int unidad_id FK
    text cambios
    text items
    text comentarios
    date fecha_alta
    date fecha_liberacion
    text estado
    boolean domiciliado
    DECIMAL latitud
    DECIMAL longitud
    boolean pendiente
    boolean reasignado_admin
}

SERVICIO_PENDIENTE {
    int id PK
    int servicio_id FK
    int usuario_id_admin FK
    int tecnico_id FK
    ENUM estado
```

```
        datetime fecha_registro
    }

SERVICIO_FALLA {
    int servicio_id FK
    int falla_id FK
}

SERVICIO_PIEZA {
    int servicio_id FK
    int pieza_id FK
    int cantidad
}

SERVICIO_HISTORIAL {
    int id_historial PK
    int servicio_id FK
    int usuario_id FK
    text accion
    datetime fecha
}

EVIDENCIA {
    int id_evidencia PK
    text url_imagen
    text descripcion
    datetime fecha
}

SERVICIO_EVIDENCIA {
    int servicio_id FK
    int evidencia_id FK
    text tipo
}

SERVICIO TIEMPO {
    int id_tiempo PK
    int servicio_id FK
    int usuario_id FK
    text accion
    datetime timestamp
}
```

```

SERVICIO_EXTENSION {
    int id_extension PK
    int servicio_id FK
    int usuario_id FK
    text duracion
    datetime fecha_solicitud
    boolean aprobado
}

%% Relaciones
ROL ||--o{ USUARIO : "tiene"
USUARIO ||--o{ SERVICIO : "levanta ticket"
USUARIO ||--o{ SERVICIO : "aprueba y devuelve"
EMPRESA ||--o{ UNIDAD : "posee"
UNIDAD ||--o{ LOG_FALLA : "registra"
FALLA ||--o{ LOG_FALLA : "registrada en"
UNIDAD ||--o{ SERVICIO : "asociada a"
SERVICIO ||--o{ SERVICIO_FALLA : "tiene"
FALLA ||--o{ SERVICIO_FALLA : "pertenece a"
SERVICIO ||--o{ SERVICIO_PIEZA : "usa"
PIEZA ||--o{ SERVICIO_PIEZA : "asignada"
SERVICIO ||--o{ SERVICIO_HISTORIAL : "registra cambios"
USUARIO ||--o{ SERVICIO_HISTORIAL : "realiza accion"
SERVICIO ||--o{ SERVICIO_EVIDENCIA : "tiene"
EVIDENCIA ||--o{ SERVICIO_EVIDENCIA : "pertenece a"
SERVICIO ||--o{ SERVICIO TIEMPO : "registra tiempo"
USUARIO ||--o{ SERVICIO TIEMPO : "accion ejecutada"
SERVICIO ||--o{ SERVICIO_EXTENSION : "tiene"
USUARIO ||--o{ SERVICIO_EXTENSION : "solicita extension"
UNIDAD ||--o{ INVENTARIO_INTERNO : "tiene inventario interno"
UNIDAD ||--o{ UNIDAD_PIEZA_REGISTRO : "tiene piezas registradas"
USUARIO ||--o{ UNIDAD_PIEZA_REGISTRO : "registrada por"
    PIEZA ||--o{ INVENTARIO_INTERNO : "registrada en inventario
interno"
    INVENTARIO_INTERNO ||--o{ UNIDAD_PIEZA_REGISTRO : "usada en
registro"
LECTORAS ||--o{ PIEZA : "opciones para"
DISCOS_DUROS ||--o{ PIEZA : "opciones para"
CABLES ||--o{ PIEZA : "opciones para"
GPS_MODELOS ||--o{ PIEZA : "opciones para"
SERVICIO ||--o{ SERVICIO_PENDIENTE : "puede quedar pendiente"
USUARIO ||--o{ SERVICIO_PENDIENTE : "admin asigna"
USUARIO ||--o{ SERVICIO_PENDIENTE : "tecnico asignado"

```

DIAGRAMA DE FLUJO

```
flowchart TD
%% =====
%% INICIO DEL FLUJO
%% =====
A[Unidad presenta falla] --> B[Registrar falla en log_fallas]
B --> C[Técnico crea servicio/ticket en servicios]

%% =====
%% EVIDENCIAS ANTES DEL SERVICIO
%% =====
C --> D[Subir evidencia inicial en evidencias]
D --> E[Técnico inicia reparación]

%% =====
%% REPARACIÓN Y EVIDENCIAS
%% =====
E --> F[Subir evidencias durante la reparación]
F --> G{¿Detener tiempo?}
G -->|Si| H[Registrar pausa en servicio_tiempo]
G -->|No| E
H --> E[Reanudar reparación - log servicio_tiempo]

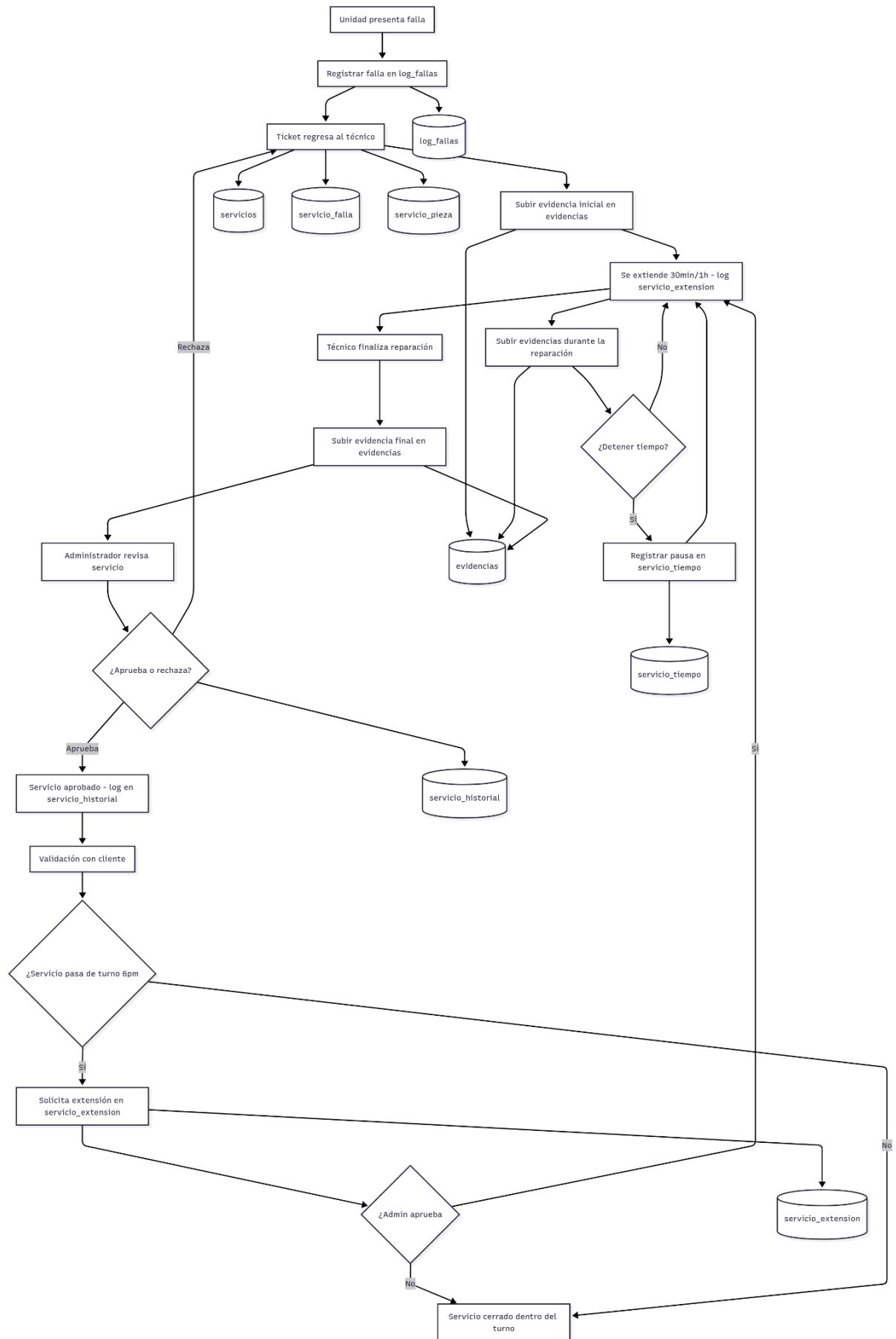
%% =====
%% FINALIZACIÓN DEL SERVICIO
%% =====
E --> I[Técnico finaliza reparación]
I --> J[Subir evidencia final en evidencias]
```

```
%% =====
%% REVISIÓN ADMINISTRADOR
%% =====
J --> K[Administrador revisa servicio]
K --> L{¿Aprueba o rechaza?}
L -->|Aprueba| M[Servicio aprobado - log en servicio_historial]
L -->|Rechaza| C[Ticket regresa al técnico]

%% =====
%% VALIDACIÓN CLIENTE
%% =====
M --> N[Validación con cliente]

%% =====
%% CONTROL DE TURNOS Y EXTENSIONES
%% =====
N --> O{¿Servicio pasa de turno 6pm}
O -->|Si| P[Solicita extensión en servicio_extension]
P --> Q{¿Admin aprueba}
Q -->|Si| E[Se extiende 30min/1h - log servicio_extension]
Q -->|No| R[Servicio cerrado a las 6pm]
O -->|No| R[Servicio cerrado dentro del turno]

%% =====
%% REFERENCIAS A BD
%% =====
B --> DB1[(log_fallas)]
C --> DB2[(servicios)]
D --> DB3[(evidencias)]
F --> DB3
J --> DB3
H --> DB4[(servicio_tiempo)]
P --> DB5[(servicio_extension)]
L --> DB6[(servicio_historial)]
C --> DB7[(servicio_falla)]
C --> DB8[(servicio_pieza)]
```



Flujo General del Sistema

1. Gestión de usuarios y roles

- Se crean **roles** (técnico, administrador, etc.).
- Se registran **usuarios** con sus datos, vinculados a un rol.
- Cada usuario inicia sesión y su actividad queda registrada (último login).

2. Gestión de empresas y unidades

- Se registran **empresas**.
- A cada empresa se le asignan **unidades** (vehículos o equipos), con sus características.

3. Inventarios y piezas

- Se define un inventario **externo (global)** de **piezas** (ligadas a opciones maestras como lectoras, discos, cables, GPS).
- Cada **unidad** tiene un **inventario interno**, que se actualiza con las piezas instaladas, su estado y cantidades.

4. Registro de cambios en unidades

- Cuando se cambia una pieza en una unidad:
 - Se actualiza el **inventario interno**.
 - Se guarda un **registro histórico (UNIDAD_PIEZA_REGISTRO)** con quién hizo el cambio, motivo, cantidad y fecha.

5. Gestión de fallas

- Se definen **tipos de fallas**.
- Se registran en **LOG_FALLA** las fallas ocurridas en una unidad, con detalle y estado (activo/inactivo).

6. Gestión de servicios (tickets de mantenimiento)

- Un usuario (técnico/administrador) **levanta un servicio (ticket)** para una unidad.
- Otro usuario (administrador) lo puede **aprobar o devolver**.
- El servicio puede tener:
 - **Fallas asociadas** (SERVICIO_FALLA).
 - **Piezas utilizadas** (SERVICIO_PIEZA).
 - **Evidencias** (imágenes, descripciones).
 - **Historial de acciones** (SERVICIO_HISTORIAL).
 - **Tiempos registrados** por cada usuario que intervino (SERVICIO_TIEMPO).
 - **Extensiones de tiempo** solicitadas y aprobadas (SERVICIO_EXTENSION).

7. Servicios pendientes y asignaciones

- Un servicio puede quedar en estado **pendiente**.
- Se registra en **SERVICIO_PENDIENTE**, donde un **admin asigna un técnico** para continuar el trabajo.

8. Finalización del servicio

1.

- El servicio se libera con fecha de cierre y estado actualizado.
- Queda todo documentado en historial, evidencias y movimientos de inventario.

Resumen

Flujo resumido en pasos

Flujo explicado con base en la BD:

1. **Falla detectada** → se guarda en `log_fallas`.
2. **Creación de ticket** → técnico abre un registro en `servicios`.
3. **Asociaciones del servicio:**
 - `servicio_falla` vincula las fallas reportadas.
 - `servicio_pieza` registra piezas usadas.
4. **Evidencias** → imágenes se guardan en `evidencias` y se relacionan en `servicio_evidencia`.
5. **Control de tiempos** → pausas/reanudaciones en `servicio_tiempo`.
6. **Finalización** → el administrador revisa, aprueba o rechaza.
 - Si aprueba → se guarda en `servicio_historial`.
 - Si rechaza → regresa al técnico.
7. **Cliente valida el servicio.**
8. **Control de turnos** → si pasa de horario, se solicita y aprueba/rechaza extensión en `servicio_extension`.

FLUJO EN INSERCIÓN DE BD

1. Insertar roles

```
INSERT INTO roles (nombre) VALUES ('Técnico'), ('Administrador');
```

2. Insertar usuarios

```
INSERT INTO usuarios (nombre, telefono, email, password, id_rol)  
VALUES  
('Carlos Pérez', '555-111-2222', 'carlos@mail.com', 'hashedpass123', 1),  
('Ana López', '555-333-4444', 'ana@mail.com', 'hashedpass456', 2);
```

3. Insertar empresas

```
INSERT INTO empresas (nombre, direccion)  
VALUES  
(Transporte Rápido SA', 'Calle Falsa 123'),  
(Logística Express', 'Avenida Siempre Viva 456');
```

5. Insertar unidades

```
INSERT INTO unidades (id_empresa, tipo, caracteristicas)  
VALUES  
(1, 'Camión de transporte', 'Capacidad 20 personas, GPS activo'),  
(2, 'Van de personal', 'Capacidad 12 personas, Aire acondicionado');
```

4. Insertar fallas

```
INSERT INTO fallas (descripcion)  
VALUES  
(Motor no arranca'),  
(Falla en sistema de frenos'),  
(GPS no funciona'),  
(Luz de tablero encendida');
```

5. Insertar piezas

```
INSERT INTO piezas (nombre, descripcion, stock)  
VALUES  
('Batería', 'Batería de 12V', 5),  
('Pastillas de freno', 'Juego completo', 10),  
('GPS', 'Dispositivo GPS', 3);
```

6. Insertar log de fallas

```
INSERT INTO servicios (id_usuario, id_unidad, cambios, items, comentarios,  
estado)  
VALUES  
(1, 1, 'Revisar motor y GPS', 'Motor, GPS', 'Urgente', 'pendiente'),  
(1, 2, 'Revisar frenos', 'Frenos', 'Prioridad alta', 'pendiente');
```

7. Crear servicios/tickets

```
INSERT INTO servicios (id_usuario, id_unidad, cambios, items, comentarios,  
estado)  
VALUES  
(1, 1, 'Revisar motor y GPS', 'Motor, GPS', 'Urgente', 'pendiente'),  
(1, 2, 'Revisar frenos', 'Frenos', 'Prioridad alta', 'pendiente');
```

8. Asociar servicios con fallas

```
INSERT INTO servicios_fallas (id_servicio, id_falla)  
VALUES  
(1, 1),  
(1, 3),  
(2, 2);
```

9. Asociar servicios con piezas

```
INSERT INTO servicios_piezas (id_servicio, id_pieza, cantidad)
VALUES
(1, 1, 1), -- batería
(1, 3, 1), -- GPS
(2, 2, 4); -- pastillas de freno
```

10. Insertar evidencias

```
INSERT INTO evidencias (id_servicio, url_imagen, descripcion, tipo)
VALUES
(1, 'https://fotos.com/motor1.jpg', 'Motor antes de revisión', 'antes'),
(1, 'https://fotos.com/gps1.jpg', 'GPS durante reparación', 'durante'),
(1, 'https://fotos.com/motor2.jpg', 'Motor reparado', 'despues');
```

11. Insertar historial de cambios

```
INSERT INTO servicios_historial (id_servicio, id_usuario, accion)
VALUES
(1, 1, 'Servicio creado'),
(1, 2, 'Administrador revisó y aprobó');
```

12. Insertar log de tiempo de servicio

```
INSERT INTO servicios_tiempo (id_servicio, id_usuario, accion)  
VALUES  
(1, 1, 'iniciar'),  
(1, 1, 'detener'),  
(1, 1, 'reactivar'),  
(1, 1, 'finalizar');
```

13. Insertar extensiones de tiempo

```
INSERT INTO servicios_extensiones (id_servicio, id_usuario, duracion,  
aprobado)  
VALUES  
(1, 1, '30min', TRUE),  
(2, 1, '1hr', FALSE);
```

Orden de inserción recomendado:

1. Roles
2. Usuarios
3. Empresas
4. Unidades
5. Fallas
6. Log de fallas
7. Servicios
8. Servicios-fallas
9. Piezas y servicios-piezas
10. Evidencias
11. Historial
12. Tiempo de servicio
13. Extensiones de tiempo

FLUJO DE PROYECTO

