

# System Zarządzania Warsztatem Samochodowym

---

**Projekt bazy danych** | MySQL 8.0+ | Styczeń 2026

---

## 1. Podstawowe założenia projektu

### 1.1 Cel projektu

Celem projektu jest stworzenie kompleksowej bazy danych dla warsztatu samochodowego, która umożliwi:

- Efektywne zarządzanie zleceniami serwisowymi
- Ewidencję klientów i ich pojazdów
- Kontrolę stanów magazynowych części zamiennych
- Śledzenie historii napraw i zmian statusów zleceń
- Generowanie raportów i analiz biznesowych
- Automatyzację procesów biznesowych poprzez procedury i wyzwalacze

### 1.2 Główne założenia

1. Wielopoziomowa obsługa użytkowników - system rozróżnia klientów i pracowników poprzez schemat dziedziczenia
2. Pełna audytowalność - każda zmiana statusu zlecenia jest rejestrowana w tabeli historii (atrybuty zmienne w czasie)
3. Integralność danych - rozbudowany system więzów CHECK, UNIQUE i kluczy obcych
4. Automatyzacja - wyzwalacze automatyzują rutynowe operacje (generowanie numerów, aktualizacja stanów)
5. Skalowalność - indeksy na kluczach obcych i kolumnach wyszukiwania zapewniają wydajność

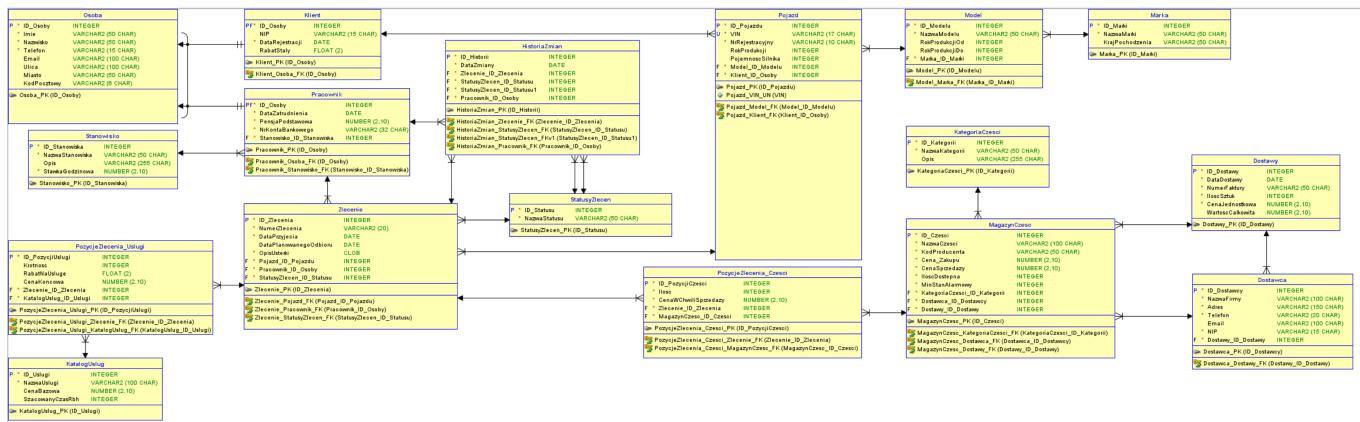
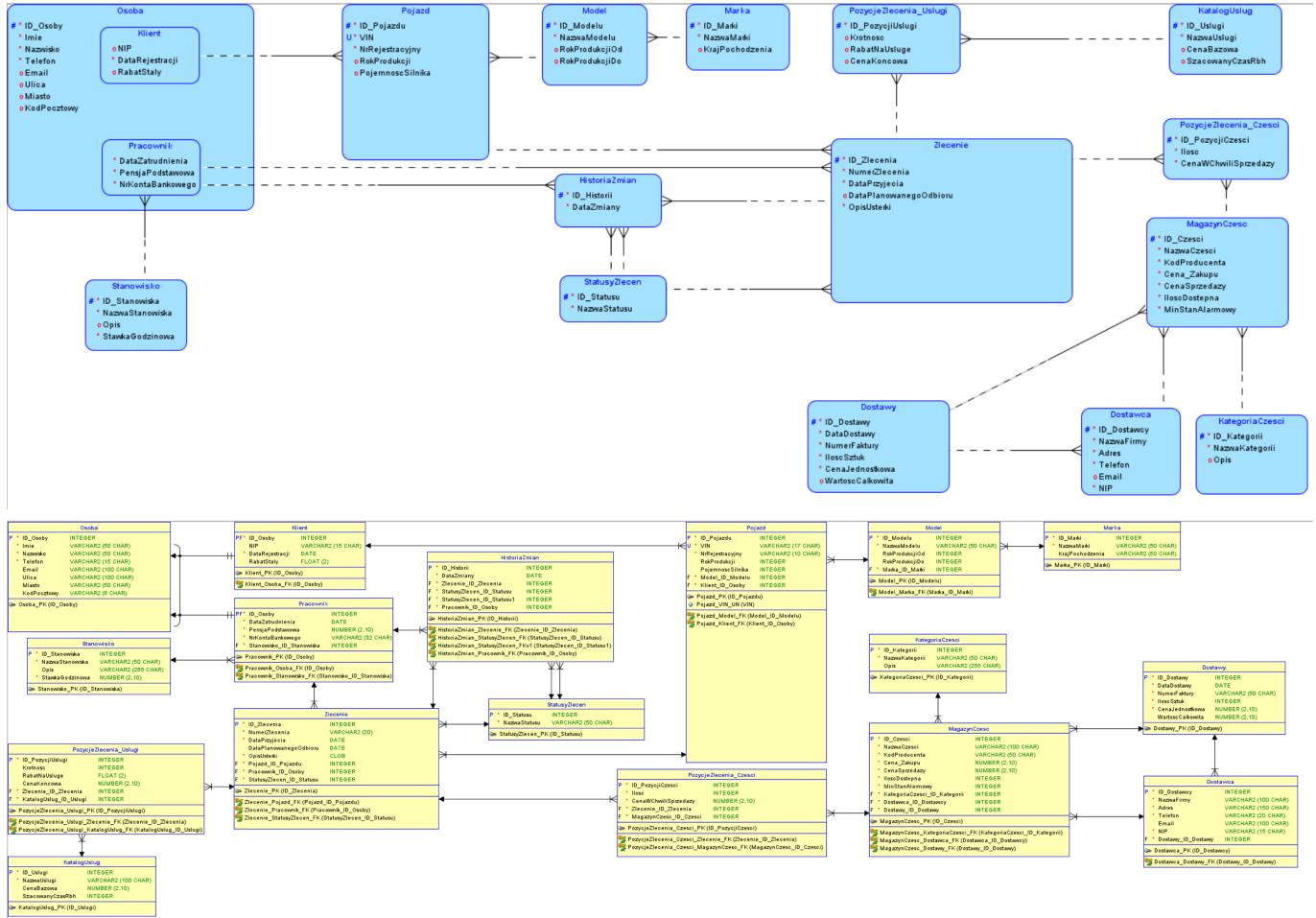
### 1.3 Możliwości systemu

- Rejestracja klientów indywidualnych i firmowych (z NIP)
- Zarządzanie flotą pojazdów klientów z pełną historią serwisową
- Obsługa zleceń serwisowych od przyjęcia do wydania pojazdu
- System rabatowy dla stałych klientów
- Kontrola stanów magazynowych z alertami niskiego stanu
- Rozliczanie usług i części z automatycznym naliczaniem rabatów
- Rejestracja dostaw od dostawców z automatyczną aktualizacją stanów
- Raporty miesięczne przychodów i statystyk

### 1.4 Ograniczenia przyjęte przy projektowaniu

- System obsługuje jeden warsztat (nie jest wielooddziałowy)
- Brak obsługi walut obcych - wszystkie ceny w PLN
- Brak integracji z systemami zewnętrznymi (ubezpieczenia, CEPiK)
- Uproszczony system uprawnień (bez ról użytkowników)
- Brak obsługi rezerwacji terminów (tylko zlecenia bieżące)

## 2. Diagram ER i Diagram Relacji



### 2.1 Lista tabel (17)

#	Tabela	Opis	Klucz główny
1	Marka	Słownik marek pojazdów	ID_Marki
2	Model	Modele pojazdów powiązane z markami	ID_Modelu
3	Stanowisko	Słownik stanowisk pracowniczych	ID_Stanowiska
4	Osoba	NADTYP - wspólne dane osobowe	ID_Osoby
5	Klient	PODTYP - dane specyficzne klientów	ID_Osoby (FK)
6	Pracownik	PODTYP - dane specyficzne pracowników	ID_Osoby (FK)
7	Pojazd	Pojazdy klientów	ID_Pojazdu
8	StatusyZlecen	Słownik statusów zleceń	ID_Statusu
9	Zlecenie	Zlecenia serwisowe	ID_Zlecenia
10	HistoriaZmian	Historia zmian statusów (atrybuty czasowe)	ID_Historii
11	KatalogUslug	Katalog dostępnych usług	ID_Uslugi
12	PozycjeZlecenia_Uslugi	Pozycje zleceń - usługi	ID_PozycjiUslugi

#	Tabela	Opis	Klucz główny
13	KategoriaCzesci	Słownik kategorii części	ID_Kategorii
14	Dostawca	Dostawcy części zamiennych	ID_Dostawcy
15	MagazynCzesc	Magazyn części zamiennych	ID_Czesci
16	PozycjeZlecenia_Czesci	Pozycje zleceń - części	ID_PozycjiCzesci
17	Dostawy	Rejestr dostaw od dostawców	ID_Dostawy

## 2.2 Klucze obce (relacje)

Tabela źródłowa	Kolumna FK	Tabela docelowa	Typ relacji
Model	ID_Marki	Marka	N:1
Klient	ID_Osoby	Osoba	1:1
Pracownik	ID_Osoby	Osoba	1:1
Pracownik	ID_Stanowiska	Stanowisko	N:1
Pojazd	ID_Modelu	Model	N:1
Pojazd	ID_Klienta	Klient	N:1
Zlecenie	ID_Pojazdu	Pojazd	N:1
Zlecenie	ID_Pracownika	Pracownik	N:1
Zlecenie	ID_AktualnegoStatusu	StatusyZlecen	N:1
HistoriaZmian	ID_Zlecenia	Zlecenie	N:1
HistoriaZmian	ID_StatusuPoprzedni	StatusyZlecen	N:1
HistoriaZmian	ID_StatusuNowy	StatusyZlecen	N:1
HistoriaZmian	ID_Pracownika	Pracownik	N:1
PozycjeZlecenia_Uslugi	ID_Zlecenia	Zlecenie	N:1
PozycjeZlecenia_Uslugi	ID_Uslugi	KatalogUslug	N:1
PozycjeZlecenia_Uslugi	ID_Pracownika	Pracownik	N:1
MagazynCzesc	ID_Kategorii	KategoriaCzesci	N:1
MagazynCzesc	ID_Dostawcy	Dostawca	N:1
PozycjeZlecenia_Czesci	ID_Zlecenia	Zlecenie	N:1
PozycjeZlecenia_Czesci	ID_Czesci	MagazynCzesc	N:1
Dostawy	ID_Czesci	MagazynCzesc	N:1
Dostawy	ID_Dostawcy	Dostawca	N:1

### 3. Dodatkowe więzy integralności danych

#### 3.1 Ograniczenia CHECK

Tabela	Ograniczenie	Opis
Marka	CHK_Marka_Nazwa	Nazwa marki min. 2 znaki
Model	CHK_Model_RokOd/Do	Rok produkcji 1886-2100
Model	CHK_Model_Lata	Rok Od ≤ Rok Do
Stanowisko	CHK_Stanowisko_Stawka	Stawka godzinowa $\geq 0$
Osoba	CHK_Osoba_Imie	Imię min. 2 znaki
Osoba	CHK_Osoba_Nazwisko	Nazwisko min. 2 znaki
Osoba	CHK_Osoba_Telefon	Format telefonu (regex)
Osoba	CHK_Osoba_Email	Format email (regex)
Osoba	CHK_Osoba_KodPocztyowy	Format XX-XXX
Klient	CHK_Klient_NIP	NIP = 10 cyfr
Klient	CHK_Klient_Rabat	Rabat 0-100%
Pracownik	CHK_Pracownik_Pensja	Pensja $\geq 0$
Pracownik	CHK_Pracownik_Konto	Nr konta = 26 cyfr
Pojazd	CHK_Pojazd_VIN	VIN = 17 znaków
Pojazd	CHK_Pojazd_Rok	Rok produkcji 1886-2100
Pojazd	CHK_Pojazd_Pojemnosc	Pojemność $> 0$
Zlecenie	CHK_Zlecenie_Koszt	Koszt $\geq 0$
KatalogUslug	CHK_Usluga_Cena	Cena $> 0$
KatalogUslug	CHK_Usluga_Czas	Czas $> 0$
KatalogUslug	CHK_Usluga_Aktywna	CzyAktywna = 'T' lub 'N'
MagazynCzesc	CHK_Magazyn_Ilosc	Ilość $\geq 0$
MagazynCzesc	CHK_Magazyn_MinStan	Min stan $\geq 0$
MagazynCzesc	CHK_Magazyn_Ceny	Cena sprzedaży $\geq$ Cena zakupu
PozycjeZlecenia_Uslugi	CHK_PozUslugi_Krotnosc	Krotność $\geq 1$
PozycjeZlecenia_Uslugi	CHK_PozUslugi_Rabat	Rabat 0-100%
PozycjeZlecenia_Czesci	CHK_PozCzesci_Ilosc	Ilość $\geq 1$
PozycjeZlecenia_Czesci	CHK_PozCzesci_Rabat	Rabat 0-100%

Tabela	Ograniczenie	Opis
Dostawy	CHK_Dostawy_Ilosc	Ilość ≥ 1

### 3.2 Ograniczenia UNIQUE

Tabela	Ograniczenie	Kolumny
Marka	UQ_Marka_Nazwa	NazwaMarki
Model	UQ_Model_Marka_Nazwa	ID_Marki, NazwaModelu
Stanowisko	UQ_Stanowisko_Nazwa	NazwaStanowiska
Pojazd	UQ_Pojazd_VIN	VIN
Pojazd	UQ_Pojazd_Rejestracja	NrRejestacyjny
StatusyZlecen	UQ_Status_Nazwa	NazwaStatusu
Zlecenie	UQ_Zlecenie_Numer	NumerZlecenia
KategoriaCzesci	UQ_Kategoria_Nazwa	NazwaKategorii
MagazynCzesc	UQ_Magazyn_Kod	KodProducenta

### 3.3 Więzy dziedziczenia

Tabele Klient i Pracownik mają klucz obcy do Osoba z opcją ON DELETE CASCADE, co zapewnia:

- Automatyczne usunięcie podtypu przy usunięciu nadtypu
- Każdy klient/pracownik MUSI mieć rekord w tabeli Osoba

## 4. Indeksy

### 4.1 Indeksy na kluczach obcych (18)

Indeksy przyspieszają operacje JOIN oraz kaskadowe usuwanie:

Indeks	Tabela	Kolumna
IDX_Model_Marka	Model	ID_Marki
IDX_Pracownik_Stanowisko	Pracownik	ID_Stanowiska
IDX_Pojazd_Model	Pojazd	ID_Modelu
IDX_Pojazd_Klient	Pojazd	ID_Klienta
IDX_Zlecenie_Pojazd	Zlecenie	ID_Pojazdu
IDX_Zlecenie_Pracownik	Zlecenie	ID_Pracownika
IDX_Zlecenie_Status	Zlecenie	ID_AktualnegoStatusu

Indeks	Tabela	Kolumna
IDX_Historia_Zlecenie	HistoriaZmian	ID_Zlecenia
IDX_Historia_Pracownik	HistoriaZmian	ID_Pracownika
IDX_Historia_StatusNowy	HistoriaZmian	ID_StatusuNowy
IDX_PozUslugi_Zlecenie	PozycjeZlecenia_Uslugi	ID_Zlecenia
IDX_PozUslugi_Usluga	PozycjeZlecenia_Uslugi	ID_Uslugi
IDX_PozUslugi_Pracownik	PozycjeZlecenia_Uslugi	ID_Pracownika
IDX_Magazyn_Kategoria	MagazynCzesc	ID_Kategorii
IDX_Magazyn_Dostawca	MagazynCzesc	ID_Dostawcy
IDX_PozCzesci_Zlecenie	PozycjeZlecenia_Czesci	ID_Zlecenia
IDX_PozCzesci_Czesc	PozycjeZlecenia_Czesci	ID_Czesci
IDX_Dostawy_Czesc	Dostawy	ID_Czesci
IDX_Dostawy_Dostawca	Dostawy	ID_Dostawcy

#### 4.2 Indeksy na kolumnach wyszukiwania (9)

Indeks	Tabela	Kolumna	Zastosowanie
IDX_Osoba_Nazwisko	Osoba	Nazwisko	Wyszukiwanie osób
IDX_Osoba_Nazwiskolmie	Osoba	Nazwisko, Imie	Wyszukiwanie kombinowane
IDX_Zlecenie_DataPrzyjecia	Zlecenie	DataPrzyjecia	Raporty, filtrowanie
IDX_Zlecenie_DataPlanowana	Zlecenie	DataPlanowanegoOdbioru	Harmonogram
IDX_Historia_DataZmiany	HistoriaZmian	DataZmiany	Audyt
IDX_Magazyn_NiskiStan	MagazynCzesc	IloscDostepna, MinStanAlarmowy	Alerty
IDX_Dostawy_Data	Dostawy	DataDostawy	Raporty dostaw
IDX_Klient_DataRejestracji	Klient	DataRejestracji	Raporty
IDX_Pracownik_DataZwolnienia	Pracownik	DataZwolnienia	Filtrowanie aktywnych

#### 4.3 Indeks na wyrażeniu (MySQL 8.0+)

Indeks	Tabela	Wyrażenie	Zastosowanie
IDX_Zlecenie_Rok	Zlecenie	(YEAR(DataPrzyjecia))	Raporty roczne

**Łącznie: 28 indeksów**

## 5. Opis widoków

### 5.1 v\_ZleceniaAktywne

**Cel:** Wyświetlenie wszystkich aktywnych zleceń (nie wydanych klientowi)

**Źródła danych:** Zlecenie, StatusyZlecen, Pojazd, Model, Marka, Klient, Osoba, Pracownik

**Kolumny:**

- ID\_Zlecenia, NumerZlecenia
- DataPrzyjecia, DataPlanowanegoOdbioru
- OpisUsterki, KosztCalkowity
- Status (nazwa)
- VIN, NrRejestracyjny, NazwaModelu, NazwaMarki
- Klient (imię i nazwisko), TelefonKlienta
- PracownikPrzyjmujacy

**Filtr:** Status NOT IN ('Wydane', 'Anulowane')

### 5.2 v\_PojazdyKlientow

**Cel:** Wyświetlenie pojazdów z danymi właścicieli

**Kolumny:** Dane pojazdu, dane właściciela, NIP, rabat, liczba zleceń

**Zastosowanie:** Wyszukiwanie pojazdów, identyfikacja klienta

### 5.3 v\_MagazynNiskiStan

**Cel:** Lista części z ilością poniżej minimalnego stanu alarmowego

**Kolumny:** Dane części, ilość dostępna, ile brakuje, dane dostawcy

**Zastosowanie:** Zamówienia uzupełniające

### 5.4 v\_PracownicyAktywni

**Cel:** Lista aktywnych pracowników ze statystykami

**Kolumny:** Dane osobowe, stanowisko, staż pracy, pensja, liczba przyjętych zleceń, liczba wykonanych usług

**Filtr:** DataZwolnienia IS NULL

### 5.5 v\_HistoriaZlecenia

**Cel:** Pełna historia zmian statusów zlecenia

**Kolumny:** Numer zlecenia, data zmiany, status poprzedni, status nowy, komentarz, kto zmienił

**Zastosowanie:** Audyt, śledzenie przepływu zlecenia

### 5.6 v\_SzczegolyZlecenia

**Cel:** Szczegółowy widok zlecenia z podsumowaniem kosztów

**Kolumny:** Wszystkie dane zlecenia, pojazdu, klienta + suma usług + suma części

**Zastosowanie:** Wydruk zlecenia, fakturowanie

## 5.7 v\_RaportMiesieczny

**Cel:** Raport miesięczny - podsumowanie zleceń

**Kolumny:** Miesiąc, liczba zleceń, zakończonych, w trakcie, przychody, średni koszt, największe zlecenie

**Grupowanie:** DATE\_FORMAT(DataPrzyjecia, '%Y-%m')

---

## 6. Opis funkcji

### 6.1 fn\_GenerujNumerZlecenia

**Sygnatura:** fn\_GenerujNumerZlecenia() RETURNS VARCHAR(20)

**Cel:** Generuje unikalny numer zlecenia w formacie ZLC/RRRR/NNNNN

**Działanie:**

1. Pobiera bieżący rok z YEAR(CURRENT\_DATE)
2. Pobiera najwyższy numer z bieżącego roku + 1
3. Zwraca sformatowany numer: 'ZLC/2026/00001'

### 6.2 fn\_ObliczWartoscZlecenia

**Sygnatura:** fn\_ObliczWartoscZlecenia(p\_id\_zlecenia INT) RETURNS DECIMAL(12,2)

**Cel:** Oblicza całkowitą wartość zlecenia

**Działanie:**

1. Sumuje CenaKoncowa z PozycjeZlecenia\_Uslugi
2. Sumuje CenaKoncowa z PozycjeZlecenia\_Czesci
3. Zwraca sumę usług + części

### 6.3 fn\_PobierzRabatKlienta

**Sygnatura:** fn\_PobierzRabatKlienta(p\_id\_pojazdu INT) RETURNS DECIMAL(5,2)

**Cel:** Pobiera rabat stałego klienta na podstawie ID pojazdu

**Działanie:**

1. Znajduje właściciela pojazdu
2. Pobiera RabatStaly z tabeli Klient
3. Zwraca rabat (0 jeśli brak)

### 6.4 fn\_SprawdzDostepnoscCzesci

**Sygnatura:** fn\_SprawdzDostepnoscCzesci(p\_id\_czesci INT, p\_wymagana\_ilosc INT) RETURNS VARCHAR(100)

**Cel:** Sprawdza czy część jest dostępna w wymaganej ilości

**Zwraca:** 'DOSTEPNA', 'BRAK - dostepne tylko X szt.' lub 'NIEZNANA CZESC'

## 7. Opis procedur składowanych

### 7.1 sp\_NoweZlecenie

**Sygnatura:**

```
sp_NoweZlecenie(
    IN p_id_pojazdu INT,
    IN p_id_pracownika INT,
    IN p_opis_usterki TEXT,
    IN p_data_planowana DATE,
    IN p_uwagi VARCHAR(1000),
    OUT p_id_zlecenia INT,
    OUT p_numer_zlecenia VARCHAR(20)
)
```

**Cel:** Tworzy nowe zlecenie serwisowe

**Działanie:**

1. Generuje numer zlecenia
2. Tworzy rekord w tabeli Zlecenie ze statusem "Nowe"
3. Dodaje pierwszy wpis do HistoriaZmian
4. Zwraca ID i numer zlecenia

### 7.2 sp\_ZmienStatusZlecenia

**Sygnatura:**

```
sp_ZmienStatusZlecenia(
    IN p_id_zlecenia INT,
    IN p_nowy_status VARCHAR(50),
    IN p_id_pracownika INT,
    IN p_komentarz VARCHAR(500)
)
```

**Cel:** Zmienia status zlecenia z automatycznym logowaniem

**Działanie:**

1. Waliduje czy nowy status jest inny niż obecny

2. Aktualizuje status w tabeli Zlecenie
3. Jeśli status = 'Wydane', ustawia DataRzeczywistegOdbioru
4. Dodaje wpis do HistoriaZmian

## 7.3 sp\_DodajUslugoDoZlecenia

### Sygnatura:

```
sp_DodajUslugoDoZlecenia(  
    IN p_id_zlecenia INT,  
    IN p_id_uslugi INT,  
    IN p_krotnosc INT,  
    IN p_id_pracownika_wyk INT,  
    IN p_rabat_dodatkowy DECIMAL(5,2)  
)
```

**Cel:** Dodaje usługę do zlecenia z automatycznym naliczeniem rabatu

### Działanie:

1. Pobiera cenę bazową usługi
2. Pobiera rabat klienta (fn\_PobierzRabatKlienta)
3. Oblicza cenę końcową z rabatem
4. Tworzy pozycję zlecenia
5. Aktualizuje KosztCalkowity zlecenia

## 7.4 sp\_DodajCzescDoZlecenia

### Sygnatura:

```
sp_DodajCzescDoZlecenia(  
    IN p_id_zlecenia INT,  
    IN p_id_czesci INT,  
    IN p_ilosc INT,  
    IN p_rabat DECIMAL(5,2)  
)
```

**Cel:** Dodaje część do zlecenia i zmniejsza stan magazynowy

### Działanie:

1. Sprawdza dostępność części
2. Pobiera rabat klienta
3. Oblicza cenę końcową
4. Tworzy pozycję zlecenia
5. Zmniejsza stan magazynowy
6. Aktualizuje KosztCalkowity zlecenia

## 7.5 sp\_RejestrujDostawe

### Sygnatura:

```
sp_RejestrujDostawe(
    IN p_id_czesci INT,
    IN p_id_dostawcy INT,
    IN p_ilosc INT,
    IN p_cena_jednostkowa DECIMAL(10,2),
    IN p_numer_faktury VARCHAR(50)
)
```

**Cel:** Rejestruje dostawę części i aktualizuje stan magazynowy

### Działanie:

1. Tworzy rekord w tabeli Dostawy
2. Zwiększa IloscDostepna w MagazynCzesc (przez trigger)
3. Opcjonalnie aktualizuje Cena\_Zakupu

## 7.6 sp\_ZamknijZlecenie

### Sygnatura:

```
sp_ZamknijZlecenie(
    IN p_id_zlecenia INT,
    IN p_id_pracownika INT
)
```

**Cel:** Zamknie zlecenie po weryfikacji kompletności

### Działanie:

1. Sprawdza czy są pozycje (usługi lub części)
2. Przelicza wartość zlecenia
3. Zmienia status na "Zakonczony"

---

## 8. Opis wyzwalaczy

### 8.1 trg\_Zlecenie\_AutoNumer

**Typ:** BEFORE INSERT na Zlecenie

**Cel:** Automatyczne generowanie numeru zlecenia

### Działanie:

- Jeśli NumerZlecenia jest NULL → generuje format ZLC/RRRR/NNNNN

## 8.2 trg\_Magazyn\_AlertNiskiStan

**Typ:** AFTER UPDATE na MagazynCzesc

**Cel:** Alert gdy stan magazynowy spada poniżej minimum

**Działanie:** Zapisuje ostrzeżenie do tabeli LogAlertyMagazyn z danymi części

**Warunek:** NEW.IloscDostepna < NEW.MinStanAlarmowy AND OLD.IloscDostepna >= OLD.MinStanAlarmowy

## 8.3 trg\_PozUslugi\_ObliczCene\_Insert / \_Update

**Typ:** BEFORE INSERT / UPDATE na PozycjeZlecenia\_Uslugi

**Cel:** Automatyczne obliczanie ceny końcowej usługi

**Formuła:** CenaKoncowa = CenaJednostkowa × Krotnosc × (1 - Rabat/100)

## 8.4 trg\_PozCzesci\_ObliczCene\_Insert / \_Update

**Typ:** BEFORE INSERT / UPDATE na PozycjeZlecenia\_Czesci

**Cel:** Automatyczne obliczanie ceny końcowej części

**Formuła:** CenaKoncowa = CenaWChwiliSprzedazy × Ilosc × (1 - Rabat/100)

## 8.5 trg\_Dostawy\_AktualizujMagazyn

**Typ:** AFTER INSERT na Dostawy

**Cel:** Automatyczna aktualizacja stanu magazynowego po dostawie

**Działanie:** Zwiększa IloscDostepna o wartość IloscSztuk z dostawy

## 8.6 trg\_Pracownik\_WalidacjaDat\_Insert / \_Update

**Typ:** BEFORE INSERT / UPDATE na Pracownik

**Cel:** Walidacja poprawności dat zatrudnienia/zwolnienia

### Reguły:

- DataZatrudnienia nie może być w przyszłości
- DataZwolnienia musi być >= DataZatrudnienia

## 9. Strategia pielęgnacji bazy danych

### 9.1 Rodzaje kopii zapasowych

Typ	Częstotliwość	Retencja	Metoda
Pełny	Codziennie 02:00	30 dni	mysqldump

Typ	Częstotliwość	Retencja	Metoda
Przyrostowy	Co 4 godziny	7 dni	Binary logs
Eksport logiczny	Tygodniowo	90 dni	mysqldump --routines

## 9.2 Skrypt backupu - pełny dump

```
#!/bin/bash
# Pełny backup bazy warsztat
DATE=$(date +%Y%m%d_%H%M%S)
BACKUP_DIR="/backup/warsztat"

mysqldump -u root -p \
    --single-transaction \
    --routines \
    --triggers \
    --events \
    warsztat > "${BACKUP_DIR}/warsztat_full_${DATE}.sql"

# Kompresja
gzip "${BACKUP_DIR}/warsztat_full_${DATE}.sql"

# Usunięcie starszych niż 30 dni
find ${BACKUP_DIR} -name "*.gz" -mtime +30 -delete
```

## 9.3 Przywracanie z backupu

```
# Przywrócenie pełnego backupu
gunzip -c warsztat_full_20260118.sql.gz | mysql -u root -p warsztat

# Przywrócenie z binary logs (point-in-time recovery)
mysqlbinlog binlog.000001 binlog.000002 | mysql -u root -p warsztat
```

## 9.4 Harmonogram (cron)

```
0 2 * * *      - Pełny backup (codziennie 02:00)
0 6,10,14,18,22 * * * - Incremental z binary logs
0 3 * * 0       - Eksport z procedurami (niedziela 03:00)
```

# 10. Typowe zapytania

## 10.1 Wyszukiwanie zlecenia po numerze rejestracyjnym

```

SELECT z.NumerZlecenia, z.DataPrzyjecia, z.OpisUsterki, s.NazwaStatusu
FROM Zlecenie z
JOIN Pojazd p ON z.ID_Pojazdu = p.ID_Pojazdu
JOIN StatusyZlecen s ON z.ID_AktualnegoStatusu = s.ID_Statusu
WHERE p.NrRejestracyjny = ?;

```

## 10.2 Lista zleceń klienta

```

SELECT z.NumerZlecenia, z.DataPrzyjecia, p.NrRejestracyjny,
       CONCAT(mk.NazwaMarki, ' ', m.NazwaModelu) AS Pojazd,
       z.KosztCalkowity, s.NazwaStatusu
FROM Zlecenie z
JOIN Pojazd p ON z.ID_Pojazdu = p.ID_Pojazdu
JOIN Model m ON p.ID_Modelu = m.ID_Modelu
JOIN Marka mk ON m.ID_Marki = mk.ID_Marki
JOIN StatusyZlecen s ON z.ID_AktualnegoStatusu = s.ID_Statusu
WHERE p.ID_Klienta = ?
ORDER BY z.DataPrzyjecia DESC;

```

## 10.3 Raport przychodów miesięcznych

```

SELECT DATE_FORMAT(DataPrzyjecia, '%Y-%m') AS Miesiac,
       COUNT(*) AS LiczbaZlecen,
       SUM(KosztCalkowity) AS Przychod,
       ROUND(AVG(KosztCalkowity), 2) AS SredniKoszt
FROM Zlecenie z
JOIN StatusyZlecen s ON z.ID_AktualnegoStatusu = s.ID_Statusu
WHERE s.NazwaStatusu = 'Wydane'
GROUP BY DATE_FORMAT(DataPrzyjecia, '%Y-%m')
ORDER BY Miesiac DESC;

```

## 10.4 Części do zamówienia (niski stan)

```

SELECT mc.NazwaCzesci, mc.KodProducenta,
       mc.IloscDostepna, mc.MinStanAlarmowy,
       (mc.MinStanAlarmowy - mc.IloscDostepna) AS DoZamowienia,
       d.NazwaFirmy AS Dostawca, d.Telefon
FROM MagazynCzesc mc
LEFT JOIN Dostawca d ON mc.ID_Dostawcy = d.ID_Dostawcy
WHERE mc.IloscDostepna < mc.MinStanAlarmowy
ORDER BY DoZamowienia DESC;

```

## 10.5 Historia zlecenia

```

SELECT hz.DataZmiany,
       sp.NazwaStatusu AS StatusPoprzedni,
       sn.NazwaStatusu AS StatusNowy,
       hz.Komentarz,
       CONCAT(o.Imie, ' ', o.Nazwisko) AS Pracownik
  FROM HistoriaZmian hz
 JOIN Zlecenie z ON hz.ID_Zlecenia = z.ID_Zlecenia
 LEFT JOIN StatusyZlecen sp ON hz.ID_StatusuPoprzedni = sp.ID_Statusu
 JOIN StatusyZlecen sn ON hz.ID_StatusuNowy = sn.ID_Statusu
 JOIN Pracownik p ON hz.ID_Pracownika = p.ID_Osoby
 JOIN Osoba o ON p.ID_Osoby = o.ID_Osoby
 WHERE z.NumerZlecenia = ?
 ORDER BY hz.DataZmiany;

```

## 10.6 Statystyki pracownika

```

SELECT CONCAT(o.Imie, ' ', o.Nazwisko) AS Pracownik,
       st.NazwaStanowiska,
       COUNT(DISTINCT z.ID_Zlecenia) AS PrzyjetychZlecen,
       COUNT(DISTINCT pu.ID_PozycjiUslugi) AS WykonanychUslug,
       IFNULL(SUM(pu.CenaKoncowa), 0) AS WartoscUslug
  FROM Pracownik p
 JOIN Osoba o ON p.ID_Osoby = o.ID_Osoby
 JOIN Stanowisko st ON p.ID_Stanowiska = st.ID_Stanowiska
 LEFT JOIN Zlecenie z ON p.ID_Osoby = z.ID_Pracownika
 LEFT JOIN PozycjeZlecenia_Uslugi pu ON p.ID_Osoby = pu.ID_Pracownika
 WHERE p.DataZwolnienia IS NULL
 GROUP BY o.Imie, o.Nazwisko, st.NazwaStanowiska
 ORDER BY WartoscUslug DESC;

```

## 10.7 Najpopularniejsze usługi

```

SELECT ku.NazwaUslugi, ku.CenaBazowa,
       COUNT(*) AS LiczbaWykonan,
       SUM(pu.CenaKoncowa) AS Przychod
  FROM PozycjeZlecenia_Uslugi pu
 JOIN KatalogUslug ku ON pu.ID_Uslugi = ku.ID_Uslugi
 GROUP BY ku.ID_Uslugi, ku.NazwaUslugi, ku.CenaBazowa
 ORDER BY LiczbaWykonan DESC
 LIMIT 10;

```

## 10.8 Wyszukiwanie klienta po nazwisku

```

SELECT o.ID_Osoby, o.Imie, o.Nazwisko, o.Telefon, o.Email,
       k.NIP, k.RabatStaly, k.DataRejestracji,
       COUNT(p.ID_Pojazdu) AS LiczbaPojazdow

```

```

FROM Osoba o
JOIN Klient k ON o.ID_Osoby = k.ID_Osoby
LEFT JOIN Pojazd p ON k.ID_Osoby = p.ID_Klienta
WHERE UPPER(o.Nazwisko) LIKE CONCAT('%', UPPER(?), '%')
GROUP BY o.ID_Osoby, o.Imie, o.Nazwisko, o.Telefon, o.Email,
        k.NIP, k.RabatStaly, k.DataRejestracji
ORDER BY o.Nazwisko, o.Imie;

```

## 11. Skrypty SQL

Wszystkie skrypty znajdują się w katalogu MySQL/:

Plik	Opis
00_INSTALL_ALL.sql	Skrypt instalacyjny (uruchamia wszystkie)
01_CREATE_DATABASE.sql	Tabele, klucze, ograniczenia, dane słownikowe
02_INDEXES.sql	Indeksy (28)
03_VIEWS_FUNCTIONS.sql	Widoki (7) i funkcje (4)
04 PROCEDURES.sql	Procedury składowane (6)
05_TRIGGERS.sql	Wyzwalacze (8)
06_TEST_DATA.sql	Dane testowe

## 12. Uruchomienie bazy danych

### Wymagania

- **Docker Desktop** - [pobierz tutaj](#)
- **PowerShell** (Windows) lub Terminal (Linux/Mac)

### Krok 1: Uruchom kontener MySQL

```

docker run -d --name mysql-warsztat -p 3306:3306 -e
MYSQL_ROOT_PASSWORD=warsztat123 -e MYSQL_DATABASE=warsztat mysql:8.0

```

### Krok 2: Poczekaj na uruchomienie bazy (~30-60 sekund)

```

# Windows PowerShell
do { Start-Sleep 3; $r = docker logs mysql-warsztat 2>&1 | Select-String "ready
for connections" } while (-not $r); "Baza gotowa!"

```

```
# Linux/Mac
while ! docker logs mysql-warsztat 2>&1 | grep -q "ready for connections"; do
sleep 3; done; echo "Baza gotowa!"
```

### Krok 3: Skopiuj pliki SQL do kontenera

```
docker cp MySQL mysql-warsztat:/tmp/
```

### Krok 4: Uruchom instalację

```
docker exec -i mysql-warsztat mysql -uroot -pwarsztat123 warsztat <
MySQL/01_CREATE_DATABASE.sql
docker exec -i mysql-warsztat mysql -uroot -pwarsztat123 warsztat <
MySQL/02_INDEXES.sql
docker exec -i mysql-warsztat mysql -uroot -pwarsztat123 warsztat <
MySQL/03_VIEWS_FUNCTIONS.sql
docker exec -i mysql-warsztat mysql -uroot -pwarsztat123 warsztat <
MySQL/04 PROCEDURES.sql
docker exec -i mysql-warsztat mysql -uroot -pwarsztat123 warsztat <
MySQL/05_TRIGGERES.sql
docker exec -i mysql-warsztat mysql -uroot -pwarsztat123 warsztat <
MySQL/06_TEST_DATA.sql
```

### Dane połączenia

Parametr	Wartość
Host	localhost
Port	3306
Baza danych	warsztat
Użytkownik	root
Hasło	warsztat123
Connection String	mysql -h localhost -P 3306 -u root -pwarsztat123 warsztat

### Połączenie z bazą przez mysql CLI

```
docker exec -it mysql-warsztat mysql -uroot -pwarsztat123 warsztat
```

### Przydatne komendy Docker

```
# Zatrzymaj kontener
docker stop mysql-warsztat

# Uruchom ponownie
docker start mysql-warsztat

# Usuń kontener (reset bazy)
docker rm -f mysql-warsztat

# Zobacz logi
docker logs mysql-warsztat
```

## Weryfikacja instalacji

Po uruchomieniu instalacji wykonaj:

```
-- Sprawdź tabele
SELECT COUNT(*) AS Tabele FROM information_schema.tables
WHERE table_schema = 'warsztat' AND table_type = 'BASE TABLE';

-- Sprawdź widoki
SELECT COUNT(*) AS Widoki FROM information_schema.views
WHERE table_schema = 'warsztat';

-- Sprawdź procedury i funkcje
SELECT routine_type, COUNT(*) FROM information_schema.routines
WHERE routine_schema = 'warsztat' GROUP BY routine_type;

-- Sprawdź triggery
SELECT COUNT(*) AS Triggery FROM information_schema.triggers
WHERE trigger_schema = 'warsztat';
```

Oczekiwany wynik:

- Tabele: 17 (+ 1 LogAlertyMagazyn)
- Widoki: 7
- Procedury: 6
- Funkcje: 4
- Triggery: 8

## 13. Podsumowanie spełnionych wymagań

<b>Wymaganie</b>	<b>Minimum</b>	<b>Zrealizowano</b>
Tabele	16 (8×2 osoby)	17
Schemat dziedziczenia	Tak	Class Table Inheritance

Wymaganie	Minimum	Zrealizowano
Atrybuty zmienne w czasie	Tak	Tabela HistoriaZmian
Widoki/funkcje	10	11 (7+4)
Procedury składowane	5	6
Wyzwalacze	5	8
Strategia backupu	Tak	mysqldump + binary logs
Indeksy	-	28
Diagram ER	Tak	Tak
Schemat relacji	Tak	Tak
Więzy integralności	Tak	~45 CHECK + UNIQUE
Typowe zapytania	Tak	8 przykładów

## 14. Autorzy

- **Karol Dziekan**
- **Krzysztof Cholewa**