

Projekt z przedmiotu „Bazy Danych”

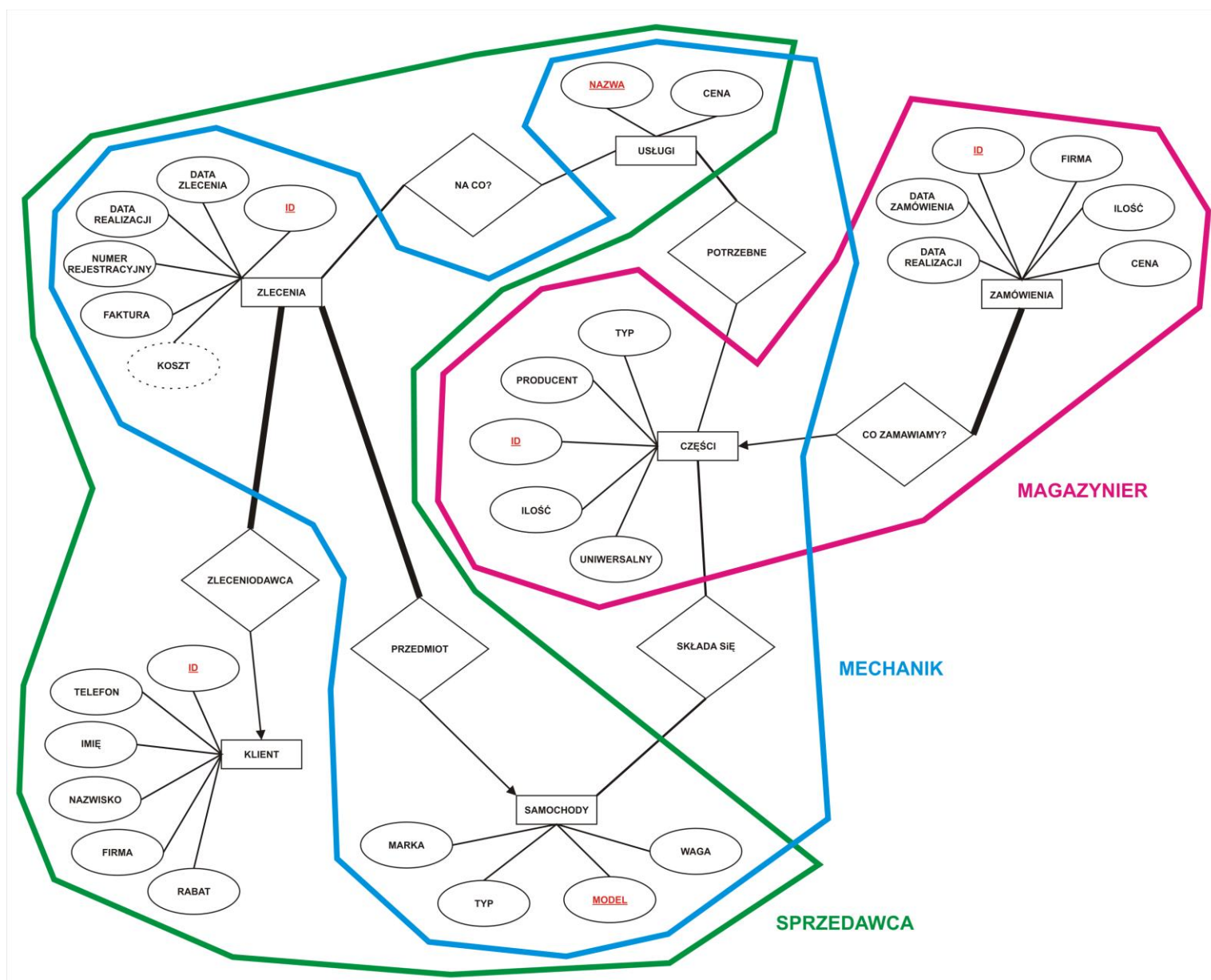
DOKUMENTACJA

Rafał Kaleta

grupa MKA

MODEL KONCEPTUALNY

DIAGRAM E-R



KOMENTARZE

- Wartość NULL w atrybucie „data realizacji” w relacjach „Zlecenia” oraz „Zamówienia” oznacza, że transakcja nie została jeszcze zrealizowana.
- Wartość NULL w atrybucie „firma” w relacji „Klient” oznacza klienta indywidualnego.
- Wartość atrybutu „uniwersalny” w relacji „Części” określa, czy część może być montowana w dowolnym modelu samochodu, określa przez to występowanie elementu w zależności „Składa się”.
- Wartości atrybutów „cena” i „koszt” są liczbami określonymi z dokładnością do dwóch miejsc po przecinku.
- Wartość atrybutu „rabat” w relacji „Klient” jest wartością całkowitą procentową przyjmującą wartości 0, 5, 10, 15, 20 lub 25.

ROLE

SPRZEDAWCA

- A. ENCJA „Zlecenia”: odczyt, dodawanie, modyfikacja
 - B. ENCJA „Klient”: odczyt, dodawanie, modyfikacja
 - C. ENCJA „Usługi”: odczyt, dodawanie, modyfikacja
 - D. ENCJA „Samochody”: odczyt
1. Oferuje usługi warsztatu samochodowego.
 2. Przyjmuje zlecenia od klientów.
 3. Dokonuje rozliczeń finansowych z klientami.
 4. Dodaje nowych klientów do bazy danych.
 5. Określa rabaty klientom.

MECHANIK

- A. ENCJA „Zlecenia”: odczyt, modyfikacja
 - B. ENCJA „Części”: odczyt, dodawanie, modyfikacja
 - C. ENCJA „Usługi”: odczyt
 - D. ENCJA „Samochody”: odczyt
1. Wykonuje zlecenia klientów.
 2. Pobiera części zamienne potrzebne do wykonania zlecenia.
 3. Informuje o zakończeniu zlecenia.

MAGAZYNIER

- A. ENCJA „Zamówienia”: odczyt, dodawanie, modyfikacja
 - B. ENCJA „Części”: odczyt
1. Monitoruje liczbę części zamiennych w warsztacie.
 2. Dokonuje zamówień części zamiennych.
 3. Dbą o odpowiedni dobór części do samochodów i usług.

OPIS DZIAŁANIA APLIKACJI

Prezentowana aplikacja umożliwia zarządzanie bazą danych warsztatu samochodowego. Została napisana w języku Python z wykorzystaniem pakietu GTK 3 oraz systemu baz danych PostgreSQL.

Aplikacja przeznaczona jest do użytku wewnętrznego warsztatu. Spełnia rolę pośrednika między pracownikiem warsztatu a bazą danych, umożliwiając mu zarządzanie wartościami przechowywanymi w bazie.

Po każdorazowym uruchomieniu aplikacji użytkownikowi ukazuje się okienko logowania do bazy danych. Pracownik wpisuje tam swój login oraz hasło. Loginy mogą składać się wyłącznie z małych i wielkich liter alfabetu angielskiego, zaś hasła wyłącznie z małych i wielkich liter alfabetu angielskiego oraz cyfr. Po pomyślnym zalogowaniu się aplikacja uruchamia okno zarządzania bazą zależne od roli, jaka została przypisana pracownikowi warsztatu. Możliwe są role *sprzedawcy*, *mechanika* oraz *magazyniera*. Każda rola posiada osobne okna dostosowane do operacji udostępnianych im przez bazę danych.

W trakcie pracy aplikacja przez okienko graficzne pobiera dane od użytkownika, które przekazuje do odpowiedniego zapytania bazodanowego. Dane wprowadzane są do pola tekstowe lub zaznaczane w polu wyboru, a aplikacja dostosowuje je do zapytania. Zatwierdzenie wprowadzonych danych odbywa się przez kliknięcie przycisku, co powoduje także pobranie tych danych z okna oraz wykonanie zapytania bazodanowego. Zwracane wyniki zapytań bądź komunikaty o błędach wyświetlane są w osobnym okienku.

Zamknięcie przez użytkownika głównego okna zarządzania powoduje przerwanie połączenia aplikacji z bazą danych oraz zakończenie działania aplikacji.

URUCHOMIENIE APLIKACJI

- 1 Wymagane oprogramowanie:
 - a. interpreter języka **Python** w wersji **2.7.x**
 - b. moduły języka Python: **psycopg2**, **gobject**
 - c. system zarządzania bazami danych **PostgreSQL**
- 2 W celu uruchomienia aplikacji
 - a. otwórz konsolę / wiersz polecenia
 - b. za pomocą polecenia **cd** przejdź do katalogu, w którym znajduje się plik **warsztat.py**, jest to główny plik aplikacji
 - c. wpisz polecenie **python warsztat.py** uruchamiające aplikację
 - d. aplikacja jest gotowa do działania; możesz zalogować się do bazy danych w okienku aplikacji

MIŁEJ PRACY!