

Firma Transportowa

Bazy danych - dokumentacja projektu

Cel projektu:

Aplikacja bazodanowa z wykorzystaniem bazy danych "Oracle 18c" i "Netbeans 8.2"

Temat projektu:

Aplikacja bazodanowa firmy transportowej z użyciem funkcjonalności zaprojektowanych w bazie danych, z interfejsem graficznym stworzonym za pomocą języka Java i biblioteki JavaFX.

Autor:

Filip Papiernik

Prowadzący:

Dr. Inż. Piotr Grochowalski

Spis Treści

Firma Transportowa	1
1. Projekt - koncepcja	4
1.1 Temat.....	4
1.2 Określenie funkcjonalności jakie spełnia aplikacja.....	4
2. Schemat ERD	6
3. Opis interfejsu	7
3.1 Panel logowania	7
3.2 Panel administratora	9
3.3 Panel pracownika	12
3.4 Panel klienta	14
3.5 Panel kuriera.....	15
4. Klasy zaimplementowane w projekcie	15
5. Przedstawienie funkcjonalności aplikacji po stronie bazy	16
6. Triggery.....	18
7. Instrukcja użytkowania.....	18
7.1 Panel Logowania i rejestrowania.....	18
7.2 Panel administratora	18

7.3 Panel pracownika	19
7.4 Panel kuriera.....	19
7.5 Panel klienta	19
8. Wymagane loginy i hasła do sprawdzenia działania aplikacji	19
8.1 Baza danych.....	19
8.2 Administrator	20
8.3 Pracownik	20
8.4 Kurier	20
8.5 Klient.....	20
9. Ograniczenia.....	20
10. Lista błędów zdefiniowanych w projekcie:.....	20

1. Projekt - koncepcja.

1.1 Temat

Tematem projektu jest aplikacja firmy transportowej. Została ona stworzona zarówno dla klientów, pracowników jak i administratorów systemu, którzy mają sprawować kontrolę nad działaniem całego systemu.

1.2 Określenie funkcjonalności jakie spełnia aplikacja.

Użytkownicy dzielą się na 4 kategorie:

- a) Administratorzy
- b) Pracownicy
- c) Klienci
- d) Kurier

Funkcjonalności dla każdego użytkownika:

- a) Administratorzy

Konta administratorów są "nad kontami" którzy mogą edytować system.

Funkcjonalności:

- dodawanie pracownika (rejestracja)
- edytowanie danych pracownika
- usuwanie pracownika z bazy danych
- edytowanie danych klienta
- dodawanie/edytowanie/usuwanie pojazdów, paczek
- zmienianie sposobu naliczania opłat np.
 - W przypadku transportu paczek: Algorytm naliczania opłat będzie brał pod uwagę rozmiar przesyłki, koszt ubezpieczenia oraz czy jest to transport krajowy czy międzynarodowy. Administrator może zmieniać stawkę za kilometr, stawkę np. za kilogram paczki i koszt ubezpieczenia oraz ewentualne naliczenie opłat za typ transportu.
 - W przypadku transportu osób: Algorytm naliczania opłat będzie brał pod uwagę ilość długość pobytu, koszt ubezpieczenia, oraz ewentualne naliczenie opłat za typ transportu (Może również za ilość osób w pojeździe i jego ewentualna pojemność). Administrator może edytować powyżej wymienione zmienne.

b) Pracownik

Konta pracowników stoją pomiędzy Klientem a Systemem.

Funkcjonalności:

- pracownik wybiera z listy klienta który chciałby zamówić transport
- wybiera typ transportu (paczki czy osób) oraz uzupełnia informacje o paczce
- uzupełnia informacje o odbiorcy paczki lub celu podróży
- wyświetla podsumowanie

c) Klient

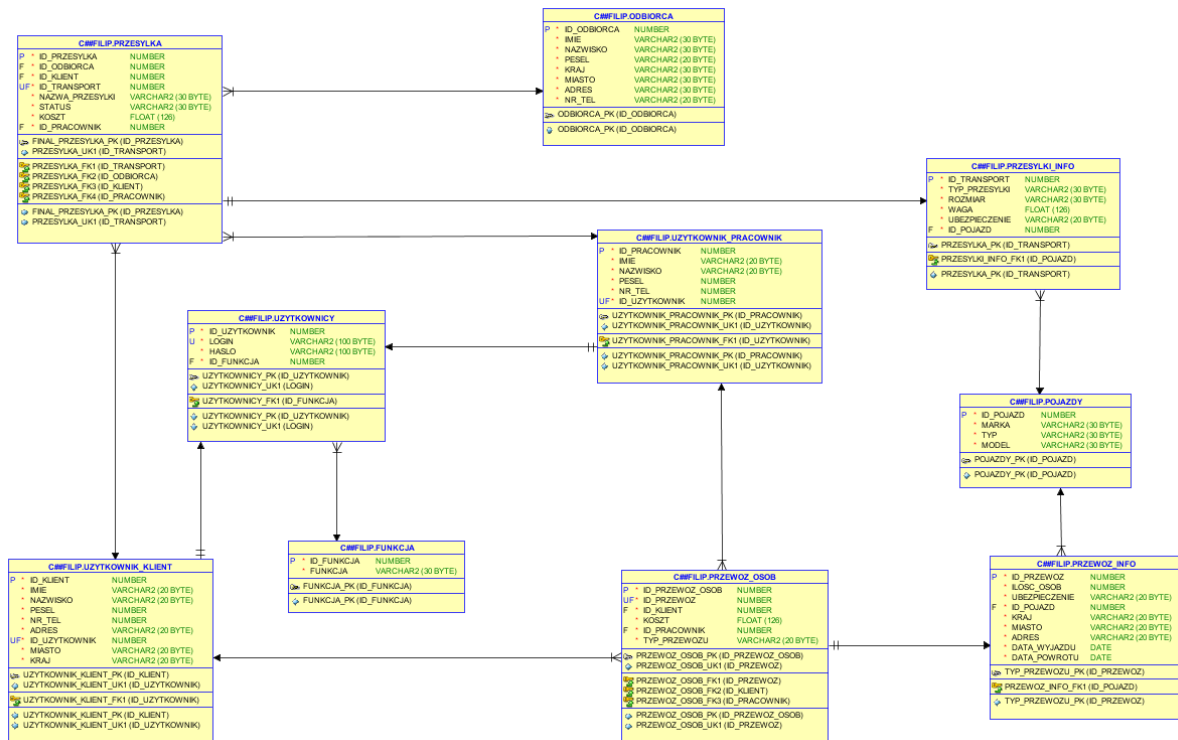
- może śledzić status aktywnych przesyłek
- może wyświetlać historie zamówionych transportów
- może wyświetlać oraz edytować swoje dane

d) Kurier

- aktualizuje status przesyłki (np. z "W transporcie" na "Dostarczono")

2. Schemat ERD

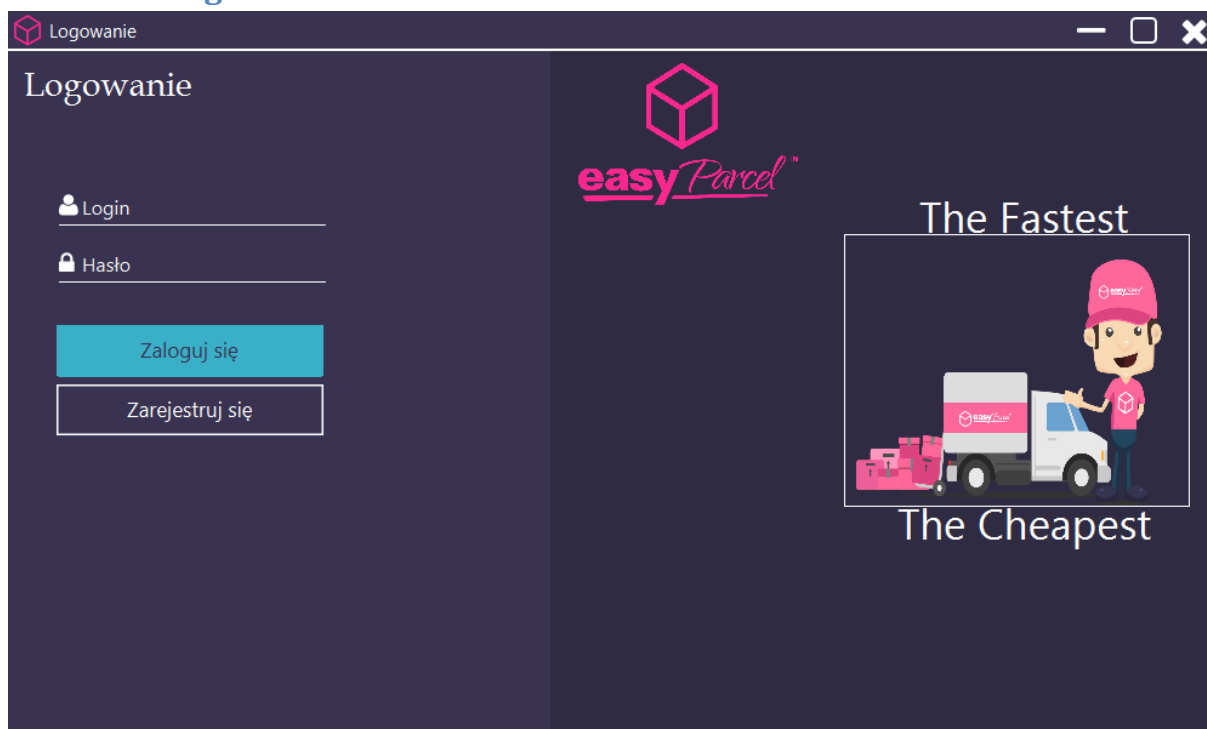
Schemat w pełnej rozdzielczości znajduje się w pliku "DiagramERD.pdf".



3. Opis interfejsu

Interfejs został zbudowany w oparciu o język Java i bibliotekę JavaFX.

3.1 Panel logowania




Rys. 1 Panel logowania.

Panel ten umożliwia zalogowanie się jako jeden z trzech rodzajów użytkowników. Widzimy też przycisk "Zarejestruj się", który umożliwia rejestrację jako klient.

Po kliknięciu przycisku "Zarejestruj się" panel z prawej strony przesuwa się i tym samym ukazują się panel rejestracji.

Rejestracja

The Fastest



The Cheapest

Rejestracja

Login

Hasło

Imię

Nazwisko

PESEL

Numer Telefonu

Kraj

Miasto

Adres

Zarejestruj

Rys 2. Panel rejestracji.

Po wypełnieniu wszystkich potrzebnych pól, przycisk Zarejestruj sprawdza ich poprawność, jeżeli wszystko jest spełnione scena zostaje przesunięta i ponownie widać panel logowania.

3.2 Panel administrativa

[illegible]

Rys 3. Panel administratora - Pracownicy.

Rys 4. Panel administratora - Pojazdy.

Panel administratora

Zalogowany jako
Filip
Papiernik
Administrator

Wyloguj

PracownicyPojazdyPaczkiTransportPrzelicznik cen

Oplaty za transport paczek

Oplata za rozmiar
5zł

Oplata za wage
3zł

Wspolne oplaty

Oplata za typ
200zł

Oplata za ubezpieczenie
50zł

Oplaty za transport osob

Oplata za jedna osobe
100zł

Oplata za dzien
300zł

Aktualizuj

Rys 7. Panel administratora - Przelicznik cen.

[illegible]

Rys 8. Panel pracownika - Wybierz klienta.

Panel Pracownika

Zalogowany jako

Filip

Papiernik

Pracownik

Wybierz Klienta

Typ i informacje

Transport paczek

Nazwa przesyłki

Rozmiar

Waga

Ubezpieczenie (T/N)

Status Rejestracja

Wybierz pojazd

Wybierz

Dalej

Wyczyść

Transport osób

Kraj

Miasto

Adres

Data wyjazdu

Data powrotu

Ilość osób

Ubezpieczenie (T/N)

Wybierz pojazd

Wybierz

Dalej

Wyczyść

Wyloguj

Rys 9. Panel pracownika - Typ i informacje.

W przyciskach 'Wybierz' znajdują się pojazdy odpowiadające tylko wybranemu typowi transportu.

Panel Pracownika

Zalogowany jako
Imie Nazwisko
Funkcja

Wybierz Klienta Typ i informacje **Dodaj odbiorce**

Wyszukaj

Imię Nazwisko
PESEL Numer telefonu
Adres
Miasto
Kraj

Dodaj nowego Wyczyść
Wybierz

Wyloguj

Rys 10. Panel pracownika - Dodaj odbiorce. (Ta scena ukazuje się tylko gdy zostanie wybrana opcja z transportem paczek)

Panel Pracownika

Zalogowany jako
Imie Nazwisko
Funkcja

Wybierz Klienta Typ i informacje Dodaj odbiorce **Podsumowanie**

Podsumowanie

=====

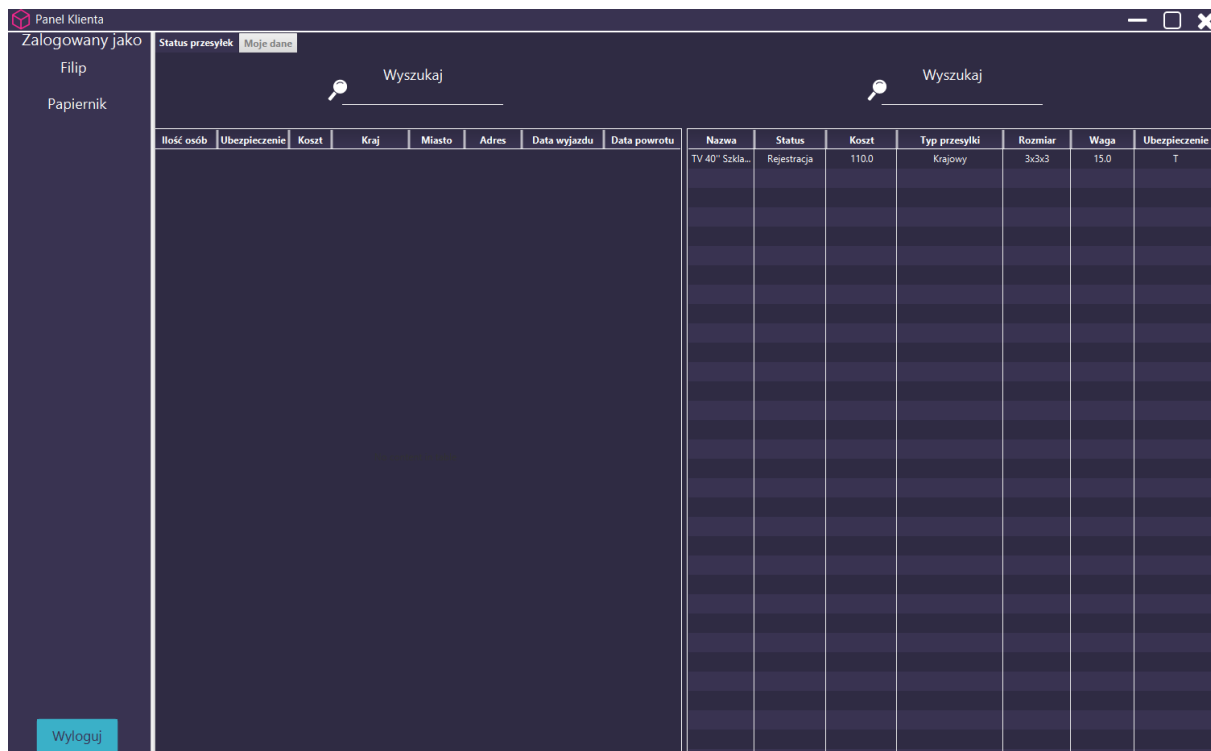
Klient	Transport
Imię:	Nazwa:
Nazwisko:	Rozmiar:
PESEL:	Waga:
Nr. Telefonu:	Ubezpieczenie:
Kraj:	Status: Dane paczki zarejestrowane
Miasto:	Pojazd:
Adres:	
Odbiorca	
Imię:	
Nazwisko:	
Pesel:	
Nr. Telefonu:	
Kraj:	
Miasto:	
Adres:	

Generuj

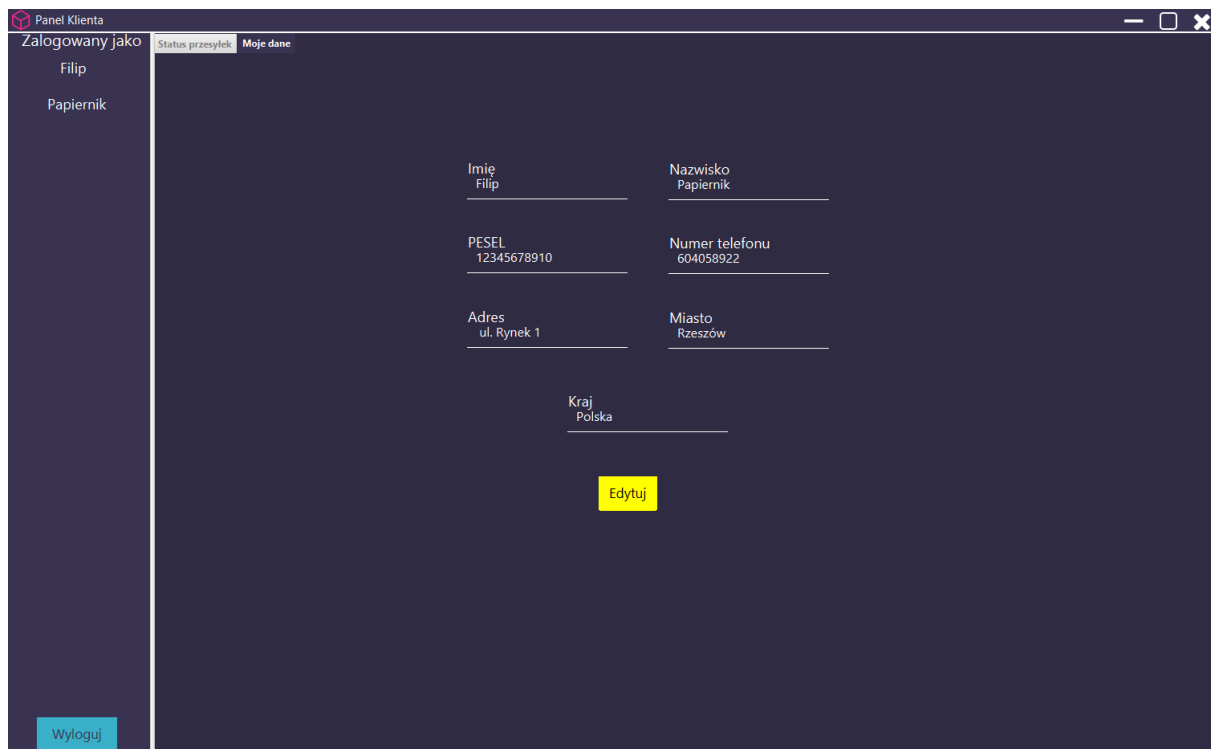
Wyloguj

Rys 11. Panel pracownika - Podsumowanie. (W zależności jaki typ został wybrany pokazuje coś innego)

3.4 Panel klienta

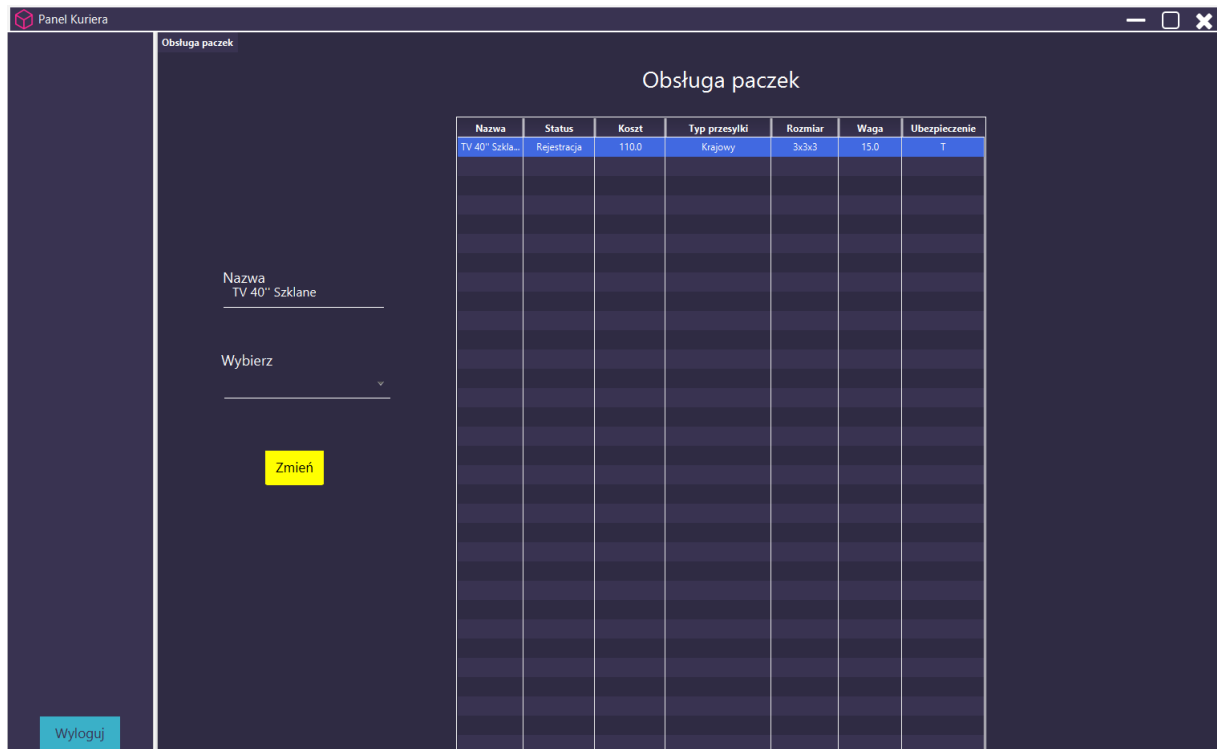


Rys 12. Panel klienta - status przesyłek.(pojawiają się tylko przesyłki z id_klienta, tzn. nie widzi on innych paczek niż swoich)



Rys 13. Panel klienta - Dane.

3.5 Panel kuriera



Rys 14. Panel kuriera - edycja statusu. (ukazują się tylko paczki jeszcze nie dostarczone)

4. Klasy zaimplementowane w projekcie

Pakiet "Entity" zawiera 11 klas reprezentujących tabele z bazy danych. (nie są zmapowane)

Każdy posiada gettery, settery i konstruktor.

Pakiet "Controllers" zawiera 5 klas, każdy do innej sceny które sterują poszczególnymi elementami GUI.

Pakiet "FXML" zawiera szablony GUI.

Pakiet "Main" zawiera dane do połączenia. (użyte raz pozwalają zmieniać je bardzo szybko)

Pakiet "CSS" zawiera pliki *.css które graficznie 'ulepszają' sceny.

5. Przedstawienie funkcjonalności aplikacji po stronie bazy

W projekcie zostały wykorzystane funkcje i procedury zaimplementowane w języku PL/SQL, znajdziemy tam wykorzystanie atrybutu %TYPE do określenia typu zmiennych wejściowych.

Przykładowe procedury:

-Wyszukiwanie pracownika

```
PROCEDURE szukaj_pracownika(u_szukane IN VARCHAR2,
                             u_wynik OUT SYS_REFCURSOR ) AS
BEGIN
OPEN u_wynik FOR
SELECT
u.*,
up.*,
f.*
FROM
UZYTKOWNICY u,
UZYTKOWNIK_PRACOWNIK up,
FUNKCJA f
WHERE u.ID_UZYTKOWNIK = up.ID_UZYTKOWNIK
AND u.id_FUNKCJA = f.ID_FUNKCJA
AND ( ( instr(upper(u.LOGIN), upper(u_szukane), 1 ) ) >0
OR ( instr(upper(up.IMIE), upper(u_szukane), 1 ) ) >0
OR ( instr(upper(up.NAZWISKO), upper(u_szukane), 1 ) ) >0
OR ( instr(upper(f.FUNKCJA), upper(u_szukane), 1 ) >0) );
NULL;
END;
```

Na rysunku powyżej znajduje się procedura która zwraca do tabeli wyszukana fraze w polu wyszukaj.

-Edytowanie przewozu

```
PROCEDURE edytuj_przewoz(przi_ilosc IN PRZEWOZ_INFO.ILOSC_OSOB%TYPE,  
                        przi_ubezpieczenie IN PRZEWOZ_INFO.UBEZPIECZENIE%TYPE,  
                        przi_kraj IN PRZEWOZ_INFO.KRAJ%TYPE,  
                        przi_miasto IN PRZEWOZ_INFO.MIASTO%TYPE,  
                        przi_adres IN PRZEWOZ_INFO.ADRES%TYPE,  
                        przi_datawyj IN PRZEWOZ_INFO.DATA_WYJAZDU%TYPE,  
                        przi_datapow IN PRZEWOZ_INFO.DATA_POWROTU%TYPE,  
                        prz_koszt IN PRZEWOZ_OSOB.KOSZT%TYPE,  
                        przi_idprzewoz IN PRZEWOZ_INFO.ID_PRZEWOZ%TYPE  
    ) AS  
V_ROWID ROWID;  
BEGIN  
    SELECT ROWID  
    INTO   V_ROWID  
    FROM   PRZEWOZ_INFO  
    WHERE  ID_PRZEWOZ = przi_idprzewoz  
    FOR UPDATE;  
  
    UPDATE PRZEWOZ_INFO  
    SET     ILOSC_OSOB = przi_ilosc, UBEZPIECZENIE = przi_ubezpieczenie, KRAJ = przi_kraj, MIASTO = przi_miasto  
    WHERE  rowid      = v_rowid;  
    COMMIT;  
  
    SELECT ROWID  
    INTO   V_ROWID  
    FROM   PRZEWOZ_OSOB  
    WHERE  ID_PRZEWOZ = przi_idprzewoz  
    FOR UPDATE;  
  
    UPDATE PRZEWOZ_OSOB  
    SET     KOSZT = prz_koszt  
    WHERE  rowid      = v_rowid;  
    COMMIT;  
  
END;
```

Na rysunku powyżej znajduje się procedura która dodaje przewóz a informacje pobiera z pól tekstowych.

6. Triggery

W projekcie użyłem 13 triggerów w celu walidacji danych.

```
create or replace TRIGGER PESEL_TRIGGER
BEFORE INSERT ON ODBIORCA
FOR EACH ROW
begin
IF length(:new.PESEL) != 11 THEN
RAISE_APPLICATION_ERROR( -20030, 'PESEL musi mieć 11 cyfr' );
end if;
end;
```

Rysunek powyżej przedstawia jeden z triggerów, służący do sprawdzania długości peselu.

```
CREATE OR REPLACE TRIGGER PESEL_EXSIT BEFORE INSERT OR UPDATE ON UZYTEKOWNIK_KLIENT
FOR EACH ROW
DECLARE
zmienna NUMBER;
BEGIN
SELECT
COUNT(*)
INTO zmienna
FROM UZYTEKOWNIK_KLIENT
WHERE UZYTEKOWNIK_KLIENT.PESEL = :new.PESEL;
IF zmienna > 0 THEN
raise_application_error(-20070, 'Taki Pesel juz istnieje!');
END IF;
END;
```

Rysunek powyżej przedstawia jeden z triggerów, służący do sprawdzenia czy istnieje już podany pesel.

7.Instrukcja użytkownika

7.1 Panel Logowania i rejestrowania

Należy wprowadzić dane, po czym nacisnąć przycisk 'Zaloguj się'. W przypadku podania niepoprawnych danych pojawi się komunikat. Gdy chcemy się zarejestrować klikamy przycisk 'Zarejestruj się' i odpowiednio się rejestrujemy (jako klient).

7.2 Panel administratora.

- W celu dodania pracownika, wypełniamy wszystkie pola oraz na końcu wybieramy jaką funkcję chcemy aby tego pracownik posiadał (np. kurier lub pracownik).
- Gdy chcemy edytować dane pracownika, wybieramy go z tabeli oraz edytujemy jego dane, następnie klikamy przycisk 'Edytuj'.
- Gdy chcemy wyczyścić pola klikamy przycisk 'Wyczyść'.
- Gdy chcemy usunąć pracownika, wybieramy go z tabeli a następnie klikamy 'Usuń'.

- W celu dodania/usunięcia/edytowania/wyczyszczenia pól w zakładce 'Pojazdy' robimy dokładnie to co powyżej.
- W celu edytowania paczki wybieramy ją z tabeli i edytujemy.
- W celu edytowania transportu wybieramy go z tabeli i edytujemy.
- W celu zaktualizowania cen, edytujemy pola w których chcemy aby cena się zmieniła i naciskamy przycisk 'Aktualizuj'

7.3 Panel pracownika

- Na samym początku musimy wybrać klienta (który uprzednio musi się zarejestrować) z tabeli i klikamy przycisk 'Wybierz'.
 - Następnie wybieramy typ transportu (paczek lub osób) i wypełniamy tam potrzebne dane, gdy wszystko wypełnimy klikamy przycisk 'Dalej'.
 - Jeżeli wybieramy 'Transport paczek' musimy dodać lub wybrać odbiorcę z listy. Gdy chcemy dodać nowego odbiorcę (którego jeszcze nie ma w bazie) to wypełniamy pola i klikamy przycisk 'Dodaj nowego'. Następnie wybieramy go z tabeli i klikamy przycisk 'Wybierz'.
 - Na samym końcu pokazuje się pole tekstowe w którym będzie potwierdzenie. Klikamy generuj gdy wszystkie dane się zgadzają.
- Gdy potwierdzenie jest wygenerowane transport zostaje dodany do bazy. Jeżeli chcemy znowu dodać następny transport klikamy przycisk 'Nowy'.

7.4 Panel kuriera

- W celu edytowania statusu paczki klikamy na nią w tabeli oraz następnie wybieramy z listy odpowiadający status i klikamy przycisk zmień. (W tabeli nie wyświetlają się zakończone przesyłki).

7.5 Panel klienta

- Na pierwszej zakładce panel wyświetla transporty.
- W celu zmiany danych przechodzimy do zakładki 'Moje dane' i tam aktualizujemy dane, następnie klikamy przycisk 'Edytuj'.

8. Wymagane loginy i hasła do sprawdzenia działania aplikacji

8.1 Baza danych

Użytkownik: C##FILIP

Sid: xe

Hasło: Filipbaza333

Zmianę tych danych należy wykonać w paczce MAIN a następnie przejść do klasy JavaPolaczenie i tam edytować funkcję ConnectorDb().

8.2 Administrator

Login: admin

Hasło: admin

8.3 Pracownik

Login: pracownik

hasło: pracownik

8.4 Kurier

Login: kurier

Hasło: kurier

8.5 Klient

Login: klient

Hasło: klient

9. Ograniczenia

- Rejestracja pracownika oraz kuriera odbywa się z konta administratora w aplikacji. (z powodów bezpieczeństwa)

10. Lista błędów zdefiniowanych w projekcie:

-20000, 'Zle dane przy logowaniu'

Odbiorca

-20030, 'PESEL musi mieć 11 cyfr'

-20031, 'PESEL moze zawierac tylko liczby'

-20032, 'NR_TEL moze zawierac tylko liczby'

-20033, 'Imie moze zawierac tylko litery'

-20034, 'Nazwisko moze zawierac tylko litery'

-20035, 'Kraj moze zawierac tylko litery'

-20036, 'Miasto moze zawierac tylko litery'

Transport

-20050, 'Data z przeszlosci'

Czy istnieje PESEL

-20070, 'Taki Pesel juz istnieje!'

Rejestracja

-20091, 'Pole imie moze miec tylko litery'

-20092, 'Pole nazwisko moze miec tylko litery'

-20093, 'Pole kraj moze miec tylko litery'

-20094, 'Pole miasto moze miec tylko litery'

-20095, 'Pesel musi miec 11 cyfr'