

Bazy Danych 1 – Dokumentacja Projektu

Temat – Sklep i Serwis Komputerowy

Michał Domin 28.01.2022

1. Projekt koncepcji, założenia

1.1. Zdefiniowanie tematu projektu

Celem projektu jest stworzenie aplikacji służącej do obsługi sklepu internetowego jak i stacjonarnego przez użytkownika, pracownika i serwisanta.

1.2. Analiza wymagań użytkownika

Użytkownik może dodawać:

- Transakcje

- Zgłoszenia serwisowe

Pracownik może dodawać:

- Transakcje

- Produkty

- Kategorie

- Magazyny

- Produkty do kategorii

- Produkty do magazynu

- Zgłoszenia serwisowe

Serwisant może dodawać:

- Naprawy

Serwisant może edytować:

- Status zgłoszenia serwisowego poprzez dokonanie naprawy

Użytkownik może wyświetlić:

- Informacje o koncie

- Kategorie

- Produkty

- Transakcje

-Zgłoszenia Serwisowe

Pracownik może wyświetlić:

-Informacje o koncie

-Kategorie

-Produkty

-Transakcje

-Magazyny

Serwisant może wyświetlić:

-Informacje o koncie

-Naprawy

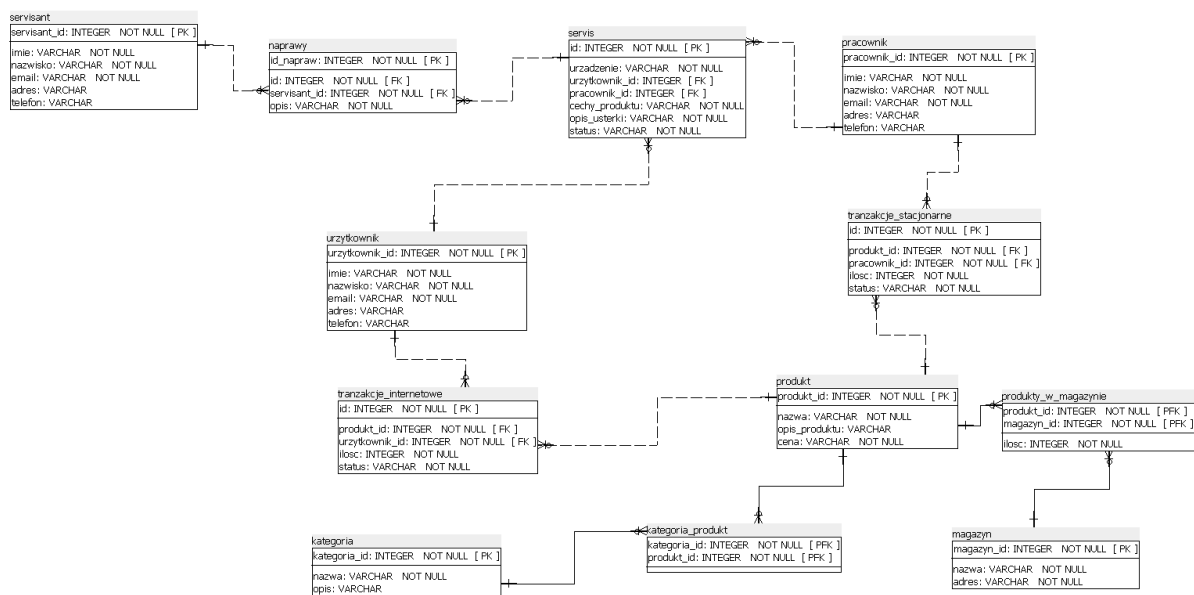
-Zgłoszenia serwisowe

1.3. Zaprojektowanie funkcji

Funkcje realizowane w bazie danych służą głównie do dodawania poprawnych danych, zapobiegania dodawania nieprawidłowych wartości, wyświetlania danych w odpowiedni sposób, oraz poprawiania danych w innych tabelach podczas wprowadzania nowych.

2. Projekt diagram

2.1. Budowa diagramu ERD



2.2. Zdefiniowanie encji

Baza zawiera następujące encje:

- 1.produkt
- 2.produkty_w_magazynie
- 3.magazyn
- 4.kategoria_produkt
- 5.kategoria
- 6.tranzakcje_internetowe
- 7.tranzakcje_stacjonarne
- 8.urzytkownik
- 9.pracownik
- 10.servis
- 11.naprawy
- 12.servisnat

2.3. Zaprojektowanie relacji pomiędzy encjami

Wszystkie relacje między tabelami to relacje 1:N

3. Projekt logiczny

3.1 Projektowanie tabel, kluczy, indeksów

Encja produkt reprezentuje pojedynczy produkt:

- produkt_id - klucz główny, generujący unikalne wartości, typu INTEGER
- nazwa – reprezentuje nazwę produktu, typu VARCHAR
- opis_produktu - reprezentuje opis produktu, typu VARCHAR
- cena – reprezentuje cenę produktu, typu VARCHAR

Encja produkty_w_magazynie reprezentuje dany produkt w magazynie:

- produkt_id – fragment klucza głównego reprezentujący produkt, typu INTEGER
- magazyn_id – fragment klucza głównego reprezentujący magazyn, typu INTEGER

-ilosc -reprezentuje ilość produktu w magazynie, typu INTEGER

Encja magazyn reprezentuje pojedynczy magazyn:

-magazyn_id - klucz główny, generujący unikalne wartości, typu INTEGER

-nazwa – reprezentuje nazwę magazynu, typu VARCHAR

-adres – reprezentuje adres magazynu, typu VARCHAR

Encja kategoria_produkt reprezentuje przynależność produktu do kategorii:

-produkt_id – fragment klucza głównego reprezentujący produkt, typu INTEGER

-kategoria_id – fragment klucza głównego reprezentujący kategorie, typu INTEGER

Encja kategoria reprezentuje pojedynczą kategorie:

-kategoria_id - klucz główny, generujący unikalne wartości, typu INTEGER

-nazwa – reprezentuje nazwę kategorii, typu VARCHAR

-opis – reprezentuje opis kategorii, typu VARCHAR

Encja tranzakcie_internetowe reprezentuje pojedynczą transakcje użytkownika:

-id - klucz główny, generujący unikalne wartości, typu INTEGER

-produkt_id –reprezentuje zakupiony produkt, typu INTEGER

-urzytkownik_id –reprezentuje użytkownika dokonującego transakcji, typu INTEGER

-ilosc – reprezentuje ilość zakupionego produktu, typu INTEGER

-status – reprezentuje status zamówienia, typu VARCHAR

Encja tranzakcie_stacjonarne reprezentuje pojedynczą transakcje pracownika:

-id - klucz główny, generujący unikalne wartości, typu INTEGER

-produkt_id –reprezentuje zakupiony produkt, typu INTEGER

-pracownik_id –reprezentuje pracownika dokonującego sprzedaży, typu INTEGER

-ilosc – reprezentuje ilość zakupionego produktu, typu INTEGER

-status – reprezentuje status zamówienia, typu VARCHAR

Encja użytkownik reprezentuje użytkownika:

- użytkownik_id - klucz główny, generujący unikalne wartości, typu INTEGER
- imie – reprezentuje imię użytkownika, typu VARCHAR
- nazwisko – reprezentuje nazwisko użytkownika, typu VARCHAR
- email – reprezentuje email użytkownika, typu VARCHAR
- adres – reprezentuje adres użytkownika, typu VARCHAR
- telefon – reprezentuje telefon użytkownika, typu VARCHAR

Encja pracownik reprezentuje pracownika:

- pracownik_id - klucz główny, generujący unikalne wartości, typu INTEGER
- imie – reprezentuje imię pracownika, typu VARCHAR
- nazwisko – reprezentuje nazwisko pracownika, typu VARCHAR
- email – reprezentuje email pracownika, typu VARCHAR
- adres – reprezentuje adres pracownika, typu VARCHAR
- telefon – reprezentuje telefon pracownika, typu VARCHAR

Encja servis reprezentuje pojedyncze zgłoszenie serwisowe:

- id - klucz główny, generujący unikalne wartości, typu INTEGER
- urządzenie – reprezentuje urządzenie poddawane serwisowi, typu VARCHAR
- użytkownik_id –reprezentuje użytkownika zgłaszającego, typu INTEGER
- pracownik_id –reprezentuje pracownika zgłaszającego, typu INTEGER
- cechu_produkta – reprezentuje cechy zgłoszonego urządzenia, typu VARCHAR
- opis_usterki – reprezentuje opis nieprawidłowości związanych z urządzeniem , typu VARCHAR
- status – reprezentuje status zgłoszenia, typu VARCHAR

Encja naprawy reprezentuje pojedynczą dokonaną naprawę na serwisowanym urządzeniu:

- id - klucz główny, generujący unikalne wartości, typu INTEGER
- id –reprezentuje naprawiane urządzenie, typu INTEGER

-servisant_id –reprezentuje serwisanta dokonującego naprawy, typu INTEGER

-opis – reprezentuje opis dokonanych napraw, typu VARCHAR

Encja servisnat reprezentuje serwisanta:

- servisnat _id - klucz główny, generujący unikalne wartości, typu INTEGER

-imie – reprezentuje imię serwisanta, typu VARCHAR

-nazwisko – reprezentuje nazwisko serwisanta, typu VARCHAR

-email – reprezentuje email serwisanta, typu VARCHAR

-adres – reprezentuje adres serwisanta, typu VARCHAR

-telefon – reprezentuje telefon serwisanta, typu VARCHAR

3.2 Słowniki danych

produkt_id	INTEGER	NOT NULL	PRIMARY KEY
nazwa	VARCHAR	NOT NULL	
opis	VARCHAR		
cena	VARCHAR	NOT NULL	

Tabela 1: Encja produkt

produkt_id	INTEGER	NOT NULL	PRIMARY KEY FOREIGN KEY
magazyn_id	INTEGER	NOT NULL	PRIMARY KEY FOREIGN KEY
ilosc	INTEGER	NOT NULL	

Tabela 2: Encja produkty_w_magazynie

magazyn_id	INTEGER	NOT NULL	PRIMARY KEY
nazwa	VARCHAR	NOT NULL	
adres	VARCHAR	NOT NULL	

Tabela 3: Encja magazyn

kategoria_id	INTEGER	NOT NULL	PRIMARY KEY FOREIGN KEY
produkt_id	INTEGER	NOT NULL	PRIMARY KEY FOREIGN KEY

Tabela 4: Encja kategoria_produkt

kategoria_id	INTEGER	NOT NULL	PRIMARY KEY
nazwa	VARCHAR	NOT NULL	
opis	VARCHAR		

Tabela 5: Encja kategoria

id	INTEGER	NOT NULL	PRIMARY KEY
produkt_id	INTEGER	NOT NULL	FOREIGN KEY
uzytkownik_id	INTEGER	NOT NULL	FOREIGN KEY
ilosc	INTEGER	NOT NULL	
status	VARCHAR	NOT NULL	

Tabela 6: Encja tranzakcie_internetowe

id	INTEGER	NOT NULL	PRIMARY KEY
produkt_id	INTEGER	NOT NULL	FOREIGN KEY
pracownik_id	INTEGER	NOT NULL	FOREIGN KEY
ilosc	INTEGER	NOT NULL	
status	VARCHAR	NOT NULL	

Tabela 7: Encja tranzakcie_stacjonarne

uzytkownik_id	INTEGER	NOT NULL	PRIMARY KEY
imie	VARCHAR	NOT NULL	
nazwisko	VARCHAR	NOT NULL	
email	VARCHAR	NOT NULL	
adres	VARCHAR		
telefon	VARCHAR		

Tabela 8: Encja uzytkownik

pracownik_id	INTEGER	NOT NULL	PRIMARY KEY
imie	VARCHAR	NOT NULL	
nazwisko	VARCHAR	NOT NULL	
email	VARCHAR	NOT NULL	
adres	VARCHAR		
telefon	VARCHAR		

Tabela 9: Encja pracownik

id	INTEGER	NOT NULL	PRIMARY KEY
urządzenie	VARCHAR	NOT NULL	
urzytkownik_id	INTEGER		FOREIGN KEY
pracownik_id	INTEGER		FOREIGN KEY
cechy_produktu	VARCHAR	NOT NULL	
opis_usterki	VARCHAR	NOT NULL	
status	VARCHAR	NOT NULL	

Tabela 10: Encja servis

id_napraw	INTEGER	NOT NULL	PRIMARY KEY
id	INTEGER	NOT NULL	
servisant_id	INTEGER	NOT NULL	FOREIGN KEY
opis	VARCHAR	NOT NULL	

Tabela 11: Encja naprawy

sevisant_id	INTEGER	NOT NULL	PRIMARY KEY
imie	VARCHAR	NOT NULL	
nazwisko	VARCHAR	NOT NULL	
email	VARCHAR	NOT NULL	
adres	VARCHAR		
telefon	VARCHAR		

Tabela 12: Encja servisnat

3.3. Analiza zależności funkcyjnych i normalizacja tabel

Tabele spełniają założenia trzeciej postaci normalnej. Wartości niekluczowych kolumn zależą od kluczy głównych. Wzajemne zależności pomiędzy kolumnami nienależącymi do klucza nie występują.

3.4 Zaprojektowanie operacji na danych

Użytkownik aplikacji może dodawać wartości do tabel poprzez formularze oraz wyświetlać raporty dotyczące danych w bazie. Raporty zasadniczo oparte są o widoki. Widoki zostały utworzone aby łatwiej można było operować na danych oraz w celu zdobycia informacji.

Widoki w bazie danych:

- lista_prod
- kategory_prod
- t_internetowe
- t_stacjonarne

Kod do widoków znajduje się w pliku 'Create_views.sql' w katalogu 'sql'.

Zaprojektowana zostały także odpowiednie wyzwalacze sprawdzające poprawność danych jak i poprawiające dane w innych tabelach przy wprowadzeniu danych:

Funkcja	Wyzwalacz	Tabela
validate_urzytkownik	urzytkownik_valid	urzytkownik
kup	kup_internet	tranzakcje_internetowe
kup	kup_stacjo	tranzakcje_stacjonarne

Tabela 13: Wyzwalacze

Kod do wyzwalaczy znajduje się w pliku 'Create_triggers.sql' w katalogu 'sql'.

Większość funkcjonalności jest napisana po stronie aplikacji w javie widoki i wyzwalacze są używane tylko do wyspecjalizowanych funkcjonalności.

4. Projekt Funkcjonalny

4.1. Interfejsy do prezentacji, edycji i obsługi danych, wizualizacja danych.

Dane możliwe do prezentacji i dodania różnią się w zależności jaki rodzaj użytkownika wybierzemy. Po lewej stronie aplikacji dla każdego użytkownika są przedstawione większość funkcjonalności których może użyć.

Większość danych wyświetlanych jest w formie tabeli opartej na TableView, część nowych danych można wprowadzać przez użycie przycisków znajdujących się w tabelach, lub poprzez kliknięcie odpowiednich opcji otwierających nowe okna z możliwością wprowadzania danych.

4.2. Zdefiniowanie panelu sterowania aplikacji

Ze sceny standardowej wybieramy jako jaki użytkownik chcemy się zalogować, na scenie logowania możemy się zalogować na już istniejące konto lub przejść na scenę rejestracji aby utworzyć nowe konto. Scena logowania i rejestracji mają opcje powrotu do poprzedniej sceny. Po zalogowaniu wyświetla się indywidualna scena dla każdego rodzaju użytkownika gdzie po lewej wypisane jest większość funkcjonalności których może użyć reszta funkcjonalności znajduje się w tabelach z przyciskami działającymi bezpośrednio na danych znajdujących się w tym samym wierszu.

5. Dokumentacja

5.1. Wprowadzenie danych

Przygotowanie bazy danych i instrukcja uruchomienia programy przedstawione są w pliku 'Readme.txt'

5.2. Dokumentacja Techniczna

Dokumentacja Techniczna aplikacji została wygenerowana za pomocą 'Javadoc' i znajduje się w katalogu 'Dokumentacja'.

5.3. Wykaz literatury:

Stackoverflow:

- <https://stackoverflow.com/>

Dokumentacja JavaFX:

- <https://docs.oracle.com/javase/8/javase-clienttechnologies.htm>

Wykłady z przedmiotu Bazy Danych 1:

- https://newton.fis.agh.edu.pl/~antek/index.php?sub=db_wykl