



**Dokumentacja projektu wykonywanego w ramach
zajęć BAZY DANYCH I**

Projekt:
Sklep Muzyczny

Majkut Piotr

I. Projekt koncepcji, założenia

1. Zdefiniowanie tematu projektu:

Celem projektu było opracowanie aplikacji bazodanowej operującej na bazie danej zapisanej w PostgreSQL na serwerze Pascal. Utworzona baza danych powinna posiadać co najmniej 7-8 tabel powiązanych związkami jak i powinna wykorzystać funkcje agregujące i widoki. Aplikacja przekazana klientowi w swoich podstawowych wymaganiach powinna pozwalać na przegląd istniejących tabel, dodawanie do nich nowych rekordów jak i wyciągnięcie co najmniej trzech raportów dotyczących poruszanej tematyki projektu. Aplikacja powinna się komunikować z klientem w języku polskim. Jako temat aplikacji obrano sklep muzyczny realizujący wysyłki online.

2. Analiza wymagań użytkownika:

Zaprojektowana baza danych sklepu muzycznego powinna umożliwić przede wszystkim utworzenie zamówienia na dany produkt z bazy danych. Produkt poza własnym opisem modelu ma opis producenta, dział oraz kategorię do której jest przypisany. Dodatkowo produkt posiada zdjęcie zakodowane za pomocą base64 wraz datą tego zdjęcia jak i historię cen wraz z aktualną. Zamówienie klienta powinno również określać kwotę i stan realizacji zamówienia. Klient poza danymi personalnymi powinien mieć możliwość zmiany adresu.

3. Zaprojektowanie funkcji:

Baza danych poza zawartością tabel umożliwia:

- sprawdzenie produktów z listy poniżej zadanej kwoty
- pokazanie informacji dotyczących statystyk sprzedaży dla dowolnego producenta (całkowita, minimalna oraz maksymalna kwota)
- sprawdzenie ilości transakcji (nie)dokonanych
- wyświetlanie historii zamówień wszystkich klientów
- sprawdzenie zamówień, dla których klient nie powinien ponosić dodatkowych kosztów za dojazd ze względu na cenę zamówienia

II. Projekt diagramów (konceptualny)

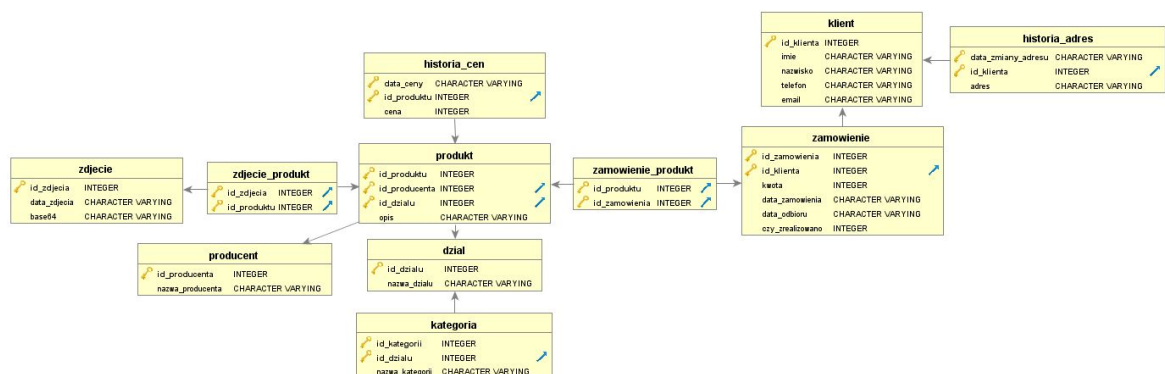
5. Zdefiniowanie encji (obiektów) oraz ich atrybutów:

Zdefiniowane podstawowe encje oraz atrybuty tych encji:

- Klient (id klienta, nazwisko, imie, adres email, nr telefonu)
- Historia zamieszkania (id klienta, data zmiany adresu, adres)
- Zamówienie (id zamówienia, id klienta, kwota, data zamówienia, data odbioru, stan realizacji)
- Produkt (id produktu, id producenta, id działu, opis produktu)
- Dział (id działu, nazwa działu)
- Kategoria (id kategorii, id działu, nazwa kategorii)
- Producent (id producenta, nazwa producenta)
- Historia ceny produktu (id produktu, cena, data nadania ceny)
- Zdjęcie (id zdjęcia, base64, data zdjęcia)

6. Zaprojektowanie relacji pomiędzy encjami:

Podgląd zdjęcia diagramu dołączonego do archiwum .zip



Adnotacja: Na etapie tworzenia bazy danych popełniono błąd, polegający na nieuwzględnieniu atrybutów np. Id_kategorii w id_dział co prowadzi do problemów z wyciągnięciem nazwy jednej kategorii dla danego produktu.

III. Projekt logiczny

7. Projektowanie tabel, kluczy, indeksów:

Do archiwum .zip został dołączony plik .sql zawierający skrypt generujący całą tablicę jak i utworzone widoki i funkcje. Klucze główne odpowiednich tabel zawierają przydomek unique, zapewniający że wszystkie wartości w kolumnie są różne od siebie.

8. Słowniki danych:

Tabela produkt poza własnym swoim własnym opisem modelu, posiada informacje o sobie przeniesione na tabele dział (opis działu), producent (nazwa producenta), kategoria (nazwa kategorii), historia cen (przechowywana jest cena produktu, która może ulec zmianie na przestrzeni czasu) oraz zdjęcie (zdjęcie zakodowane jest za pomocą base64 wraz z datą nadania).

Tabela klient poza danymi personalnymi ma dołączoną relację historia adresów, gdzie zapisane są informacje o miejscu zamieszkania

<https://www.base64encode.net/base64-image-encoder>)

Ze względu na istniejące tylko dwa stany realizacji zamówienia, nie wprowadzono dodatkowej tabeli opisującej poszczególne stany zamówienia.

9. Analiza zależności funkcyjnych i normalizacja tabel (dekompozycja do 3NF ewentualnie BCNF):

- Baza danych spełnia pierwszą postać normalną (wartości atrybutów są niepodzielne, baza nie zawiera powtarzających się grup informacji, a kolejność wierszy może być dowolna
- Druga postać normalna: przykład tabel produkt-dział-kategoria pokazują, że żadna kolumna nie kluczowa nie jest częściowo funkcyjnie zależna od jakiegokolwiek klucza potencjalnego.
- Trzecia postać normalna: tabela zamówienie zawiera informację o dacie odbioru i stanie realizacji. W tym przypadku można by jednak przyjąć, że data odbioru wskazuje na potwierdzenie realizacji zamówienia, co wskazuje na niepotrzebny natłok informacji przy tylko dwóch stanach realizacji. Dlatego ta relacja nie spełnia trzeciej postaci normalnej

10. Denormalizacja struktury tabel:

11. Zaprojektowanie operacji na danych:

Do archiwum .zip dołączono plik .sql zawierający skrypt generujący całą tablicę jak i utworzone widoki i funkcje.

Zdefiniowane dodatkowe widoki i funkcje w bazie danych

- `ilosc_przeprowadzonych_transakcji(stan_realizacji int)`: zwraca liczbę przeprowadzonych lub niezrealizowanych transakcji
- `producent_statystyki_sprzedazy(opis_producenta varchar)`: zwraca statystyki sprzedaży dla danego producenta
- `produkty_ponizej_danej_kwoty(zadana_kwota int)`

Zwraca liczbę produktów w sklepie poniżej zadanej kwoty

- Historia_Zamowien

Ukazuje tablice z danymi o wszystkich zamówieniach

- Zamowienia_z_darmowym_dojazdem

Zwraca listę zamówień, dla których dojazd do klienta jest gratis

IV. Projekt funkcjonalny

12. Interfejsy do prezentacji, edycji i obsługi danych:

Formularze do wprowadzania nowych rekordów wyświetlają się po wyborze z listy odpowiedniej tabeli. Każdy z formularzy ma przyszykowane pola, wraz z opisami oraz oczekiwanymi wartościami. Uzupełnienie wszystkich pól jest obowiązkowe, podanie złej wartości może spowodować błąd aplikacji. Zamknięcie okna formularza zwróci użytkownika do menu głównego

Dodanie rekordu do bazy

id_klienta: int

imie: String

nazwisko: String

telefon: String

email: String

Wyślij dane

W przypadku raportu może pojawić się oczekiwany zbiór wartości, z jakiego oczekiwana jest wartość. Przy wywołaniu danej funkcji posiadającej cyfrę w nawiasie, wszystkie pola oznaczone tą samą cyfrą muszą być wypełnione odpowiednimi wartościami.

Wybór raportu

(1) Stan realizacji zamówienia: {0,1}

(2) Kwota: int

(2) Producent: String

(1) Ilość (nie)przeprowadzonych transakcji

(2) Produkty poniżej danej kwoty

(3) Producent statystyki sprzedaży

Historia zamówień

13. Wizualizacja danych:

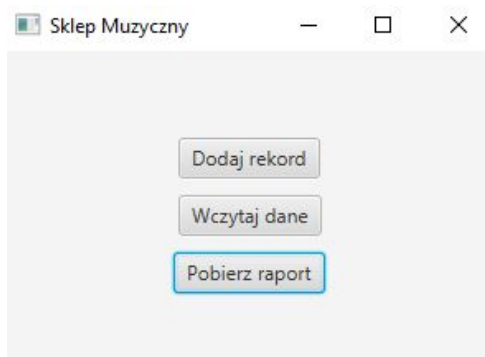
Raporty w aplikacji stanowią tabelę utworzoną na podstawie widoków lub wywołań funkcji z bazy danych zwracającą. W przypadku funkcji użytkownik może generować wiele raportów lub statystyk w zależności od podanych argumentów. Poniżej widnieje przykład rezultatu zwracanego przez raport w postaci tabeli.

Zawartosc tablicy

id_zamowienia	data_zamowienia	nazwisko	imie	email	telefon	
3	10/10/18	Wegner	Conrad	Wegner@gmail.com	685432992	
4	12/01/19	Tonye	Simon	Tonye@gmail.com	555222321	
6	12/01/19	Ericson	Steven	Ericson@gmail.com	999777333	
2	12/11/18	Ericson	Steven	Ericson@gmail.com	999777333	
7	13/01/19	Wegner	Conrad	Wegner@gmail.com	685432992	
5	15/01/19	Tonye	Simon	Tonye@gmail.com	555222321	
8	19/01/19	Tonye	Simon	Tonye@gmail.com	555222321	
1	31/12/18	Ericson	Steven	Ericson@gmail.com	999777333	

14. Zdefiniowanie panelu sterowania aplikacji:

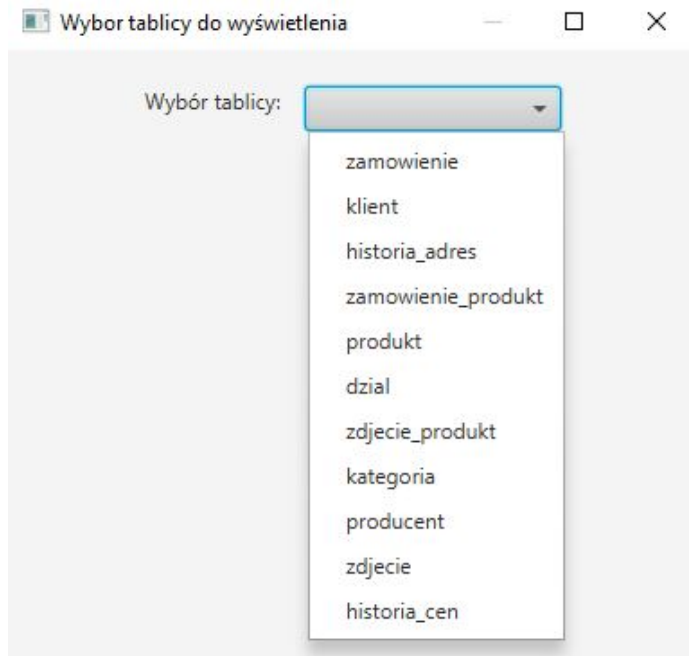
Menu główne:



- Dodaje rekord: przekierowanie do okna wyboru tablicy przed pojawieniem się formularza
- Wczytaj dane: przekierowanie do okna wyboru tablicy do wyświetlenia
- Pobierz raport: przekierowanie do okna z generowaniem raportów widocznego w podpunkcie 12. (drugie zdjęcie)

Okno wyboru: pojawia się po wybraniu opcji “Dodaj rekord” lub “Wczytaj dane”.

Wybranie nazwy tabeli przekieruje do okna z formularzem (przykład: podpunkt 12. zdjęcie nr 1) lub wygeneruje tablicę z prezentującą wszystkie rekordy (format taki jak na zdjęciu w podpunkcie 13).



15. Makropolecenia:

- Naciśnięcie przycisku "Wyślij dane" w oknie formularza spowoduje wyświetlenie informacji o (nie)pomyślnym wysłaniu danych do bazy.

V. Dokumentacja

16. Wprowadzanie danych:

- dane do aplikacji wprowadzane są ręczne
- użytkownik wprowadza tylko poszczególne dane dla nowych rekordów lub przekazuje parametry przy generowaniu raportów

17. Dokumentacja użytkownika:

- Wybranie dowolnej opcji z menu głównego spowoduje utworzenie nowego okna, menu pozostaje otwarte w tle, jednakże priorytet uzyskują nowo tworzone okna
- Zamknięcie dowolnego okna (poza głównym menu) powoduje powrót do głównego okna, zamykając po drodze okna wyboru tablic
- Pola formularzy powinny być wypełniane zgodnie z podpowiedziami widniejącymi w polu tekstowym

- Jeśli dany raport posiada cyfrę w nawiasie, to wszystkie pola oznaczone tą samą cyfrą powinny być wypełnione odpowiednią wartością

18. Opracowanie dokumentacji technicznej:

Do archiwum .zip dołączono folder 0.Projekt_Dok wraz z dokumentacją wygenerowaną przez JavaDoc.

19. Wykaz literatury:

- <https://www.youtube.com/watch?v=FLkOX4Eez6o&list=PL6gx4Cwl9DGBzfXLWLSYVy8EbTdpGbUIG>
- <http://www.postgresqltutorial.com/postgresql-jdbc/?fbclid=IwAR0Fb2GZ3XaR2OM4TgBI0X8yLclXCZKEHvJpMpsKGf-A1N8kfrArAz-u3wA>
- https://www.journaldev.com/235/java-mysql-ssh-jsch-jdbc?fbclid=IwAR1tJ0mZL1R_8zN-pHTeJMgR-AjDj7aTqhCtL7rwM839dkmS2OGvXSVwW4I