





Uniwersytet Gdański Wydział Matematyki, Fizyki i Informatyki Instytut Informatyki

Wirtualna uczelnia

Damian Shroder

Projekt z przedmiotu bazy danych na kierunku informatyka profil ogólnoakademicki na Uniwersytecie Gdańskim.



Gdańsk 26 maja 2020

Spis treści

1	Wp	Wprowadzenie						
2	Opis projektu							
	2.1	Potencjalne grupy użytkowników	2					
	2.2	Wymagania funkcjonalne	3					
	2.3	Wymagania niefunkcjonalne	3					
	2.4	Diagram zwiazków encji	5					
3	Przykłady realizacji bazy danych							
	3.1	Przykłady zawartości najważniejszych tabel	6					
	3.2	Przykłady kilku zapytań i ich wyników	8					

1 Wprowadzenie

Baza danych przeznaczona jest dla uczelni. Znajduja sie w niej takie informacje <mark>o</mark> wydziałach, kierunkach, studentach, klasach, nauczycielach, kursach, i wiele wiecej.

Podstawowe pojecia:

Baza danych - zorganizoweny zbiór informacji zawierajacy jednolity rodzaj danych,

 ${\bf Rekord}$ - pojedyńczy wiersz w tabeli, który zawiera informacje dotyczace określonego elementu,

Pole - pojedyńcza kolumna w tabelii, która określa rodzaj przechowywanych w niej informacji,

Klucz podstawowy - pojedyńcze pole, którego wartości w danej tabeli sa unikatowe, identyfikuje on poszczególne elementy,

Klucz obcy - pole, które jest kluczem podstawowym w innej tabeli,

Kwerenda - zapytania, które maja na celu powstanie tabel wirtualnych, sa tworzone na chwile, nie zapisuja sie,

Formularz - interfejs graficzny do wprowadzania danych do tabeli,

Raport - służy do wyświetlania danych z tabel i/lub kwerend w sposób przygotowany do wydruku,

Model relacyjny - istnieje kilka tabel, które sa połaczone relacjami,

System baz danych - skomputeryzowany system przechowywania danych, zorganizowany w pliku,

Encja - każdy przedmiot, zjawisko, stan, pojecie, obiekt, który potrafimy odróznić od innych obiektów,

Atrybuty - własności encji,

Zwiazek - nazwana zależność pomiedzy podstawowymi zbiorami encji,

Normalizacja - proces projektowania baz danych, tak aby utworzyć zbiór tabel o odpowiedniej strukturze.

2 Opis projektu

Projekt powstał na zlecenie rady szkolnej Uniwersytetu Bolesławskiego. Jest to cześć wiekszego projektu, który ma na celu zmodernizowanie obecnego systemu szkolnego. Zlecene zadania zostało wykonane zgodnie z najwyższymi normami.

2.1 Potencjalne grupy użytkowników

Potencjalni użytkownicy bazy danych to:

• Administrator - główny zarzadca bazy danych, posiada pełen dostep do bazy danych

- Informatyk posiada wiekszość uprawnień, dzieki którym nie trzeba za każdym razem wzywać głównego administratora
- Nauczyciel posiada dostep do bazy studentów, listy obecności, przedmitów jakich naucza, kursów oraz edycji ocen
- Student może sprawdzić przydział klas, swoje dane, liste przedmiotów, kursów, na które uczeszcza oraz wglad w oceny

2.2 Wymagania funkcjonalne

Nasza baza danych bedzie posiadała informacje o studentach, nauczycielach, ich tytule/stopniu, przedmiotach, kursach, ich poziomie, klasach, lekcjach i obecności na nich, wydziałach, kierunkach.

Dostepny jest proces złożenia pracy dyplomowej, dodania wielu adresów - także klas, kursów, przedmiotów, ocen - dla jednej osoby.

Wprowadzono zabezpieczenia m.in. do takich pól jak:

- Numer telefonu,
- Kod pocztowy,

Zautomatyzowano m.in. nastepujace pola:

- Email wpisywany jest na podstawie pierwszej litery imienia oraz nazwiska,
- Data modyfikacji jest aktualizowana przy dowolnej zmianie rekordu,
- Wynik w procesie składania pracy dyplomowej jest obliczany na podstawie pomniejszych ocen,
- Data obronienia pracy dyplomowej jest wpisywana automatycznie, jeżeli obrona jest pozytywna (zaliczona).

Dla ułatwienia wpisywania danych zastosowano formularze, w których można bezpośrednio wprowadzać studentów, nauczycieli, wystawiać oceny czy przypisywać przedmioty.

2.3 Wymagania niefunkcjonalne

Baza danych jest postawiona w chmurze [1]Azure, zaimportowana z [2]Microsoft Access.

Używany standard jezyka SQL: ANSI-89 Level 1.

Zalety:

- Pozwala na definiowanie parametrów przy definiowaniu kwerend,
- Dostarcza dodatkowych funkcji agregujacyh, jak StDev czy VarP,

Wady:

- Brak niekótrych funkcji, np. SUM czy LIMIT,
- Składnia niektórych poleceń może sie nie co różnić.

Microsoft Acces pozwala na:

- Projektowanie aplikacji w jezyku 4GL,
- Umożliwia automatyzacje pracy,
- Posiada filtry do wiekszości użytecznych formatów,
- Wchodzi w skład Microssoft Office.

Z kolei jego wady:

- Trudno oddzielić kod aplikacji od danych, brak możliwości kompilacji kodu,
- Cieżko, a czasami jest to nie możliwe, przenieść na inne platformy,

Microsoft Azure - zalety:

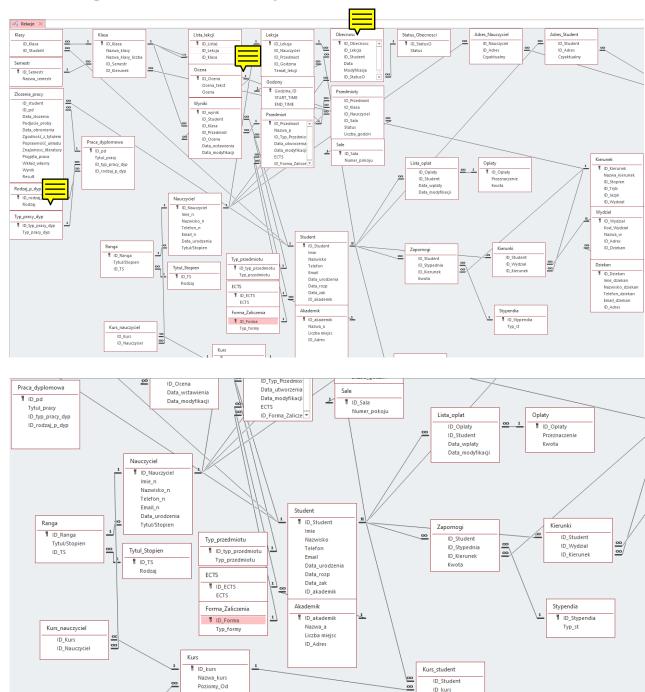
- Wysoka dostepność i mała awaryjność,
- Dobre zabezpieczenia,
- Wysoka skalowalność,
- Możliwość dostosowywania interfejsu,
- Wysoki stosunek ceny do jakośći.

Microsoft Azure - minusy:

• Wymaga dużej ilości wiedzy i doświadczenia, by sprawnie sie poruszać.



2.4 Diagram zwiazków encji



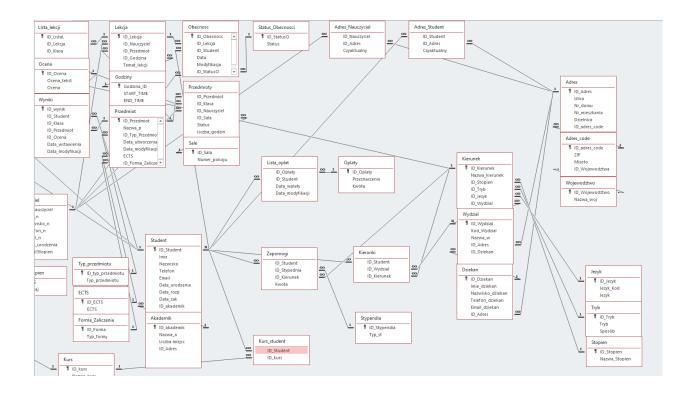
Poziomy_Do Cena_kurs Liczba_godzin

Limit_uczestnike

Data_rozp Data_zak

Poziom

ID_Poziom



3 Przykłady realizacji bazy danych

W pierwszej tabeli, dwa pola zostały zmodyfikowane:

- Telefon zawiera maske wprowadzania, która zapobiega wprowadzaniu niechianych znaków,
- Email generowany automatycznie na podstawie pierwszej litery imienia oraz nazwiska.

Pierwsze zapytanie zwraca studentów, klasy do których uczeszczaja oraz przedmioty, jakie maja przypisane w danej klasie.

Drugie natomiast zwraca średnia ocen z podziałem na klasy, do których sa przypisani.

3.1 Przykłady zawartości najważniejszych tabel

Przykładowa tabele sql zamieszczamy przy pomocy polecenia sqltable:

Field	+ Type	Null	Key	Default	Extra
ID_STUDENT IMIE NAZWISKO	+ auto_inc varchar(255) varchar(255) varchar(255)	NO NO NO	PRI 	None None None	

DATA_URO	date	+			yes
++ Field	Туре			Default	•
ID_OBECNOSC ID_LEKCJA ID_STUDENT DATA MODYFIKACJA ID_STATUSO ID_NAUCZY	int int data data int int	NO NO NO NO NO NO	 FOR FOR	None None None	
Field				Default	•
KOD_WY	auto_inc varchar(255) varchar(255) int int		PRI FOR	None None	

ID_PRZEDMI auto_inc	Field	+ Type	Null	Key	Default	Extra
ID_TYP_PRZ int	ID_PRZEDMI	auto_inc	l NO	PRI	None	Ī
DATA_UTW	ID_TYP_PRZ	int	l NO	FOR	None	Ī
ECTS						
	_			•		
	•		· ·			İ

3.2 Przykłady kilku zapytań i ich wyników

Przykłady umieszczamy przy użyciu specjalnych narzedzi do wstawiania kodu, a dokładniej pakietu lstlisting.

```
SELECT Student.Imie, Student.Nazwisko, Klasa.Nazwa_klasy, Klasa.
01 |
        ID_Semestr, Klasa.ID_Kierunek, Przedmiot.Nazwa_p, Przedmiot.
        ID_Typ_Przedmiotu
    FROM Student INNER JOIN (Przedmiot INNER JOIN ((Klasa INNER JOIN Klasy
        ON Klasa.ID_Klasa = Klasy.ID_Klasa) INNER JOIN Przedmioty ON Klasa.
        ID_Klasa = Przedmioty.ID_Klasa) ON Przedmiot.ID_Przedmiot =
        Przedmioty.ID_Przedmiot) ON Student.ID_Student = Klasy.ID_Student
    GROUP BY Student. ID_Student, Student. Imie, Student. Nazwisko, Klasa.
03 |
        ID_Klasa, Klasa.Nazwa_klasy, Klasa.ID_Semestr, Klasa.ID_Kierunek,
        Przedmiot.ID_Przedmiot, Przedmiot.Nazwa_p, Przedmiot.
        ID_Typ_Przedmiotu;
04 |
     SELECT Student.Imie, Student.Nazwisko, Klasa.Nazwa_klasy, Klasa.
        ID_Semestr, Klasa.ID_Kierunek, Avg(Ocena.Ocena) AS redniaOfOcena
     FROM Student INNER JOIN (Przedmiot INNER JOIN (Ocena INNER JOIN (Klasa
        INNER JOIN Wyniki ON Klasa.ID_Klasa = Wyniki.ID_Klasa) ON Ocena.
        ID_Ocena = Wyniki.ID_Ocena) ON Przedmiot.ID_Przedmiot = Wyniki.
        ID_Przedmiot) ON Student.ID_Student = Wyniki.ID_Student
     GROUP BY Student. ID_Student, Student. Imie, Student. Nazwisko, Klasa.
        ID_Klasa, Klasa.Nazwa_klasy, Klasa.ID_Semestr, Klasa.ID_Kierunek;
```

Literatura

- [1] Microsoft, Azure, 2017.
- [2] _____, Jezyk access sql: pod owe pojecia, słownictwo i składnia, 2019.