



Uniwersytet Gdański  
Wydział Matematyki, Fizyki i Informatyki  
Instytut Informatyki

# Baza danych dla serwisu z kursami online

Paweł Bukowski



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

Gdańsk  
23 maja 2020

## Spis treści

<b>1</b>	<b>Wprowadzenie</b>	<b>2</b>
<b>2</b>	<b>Opis projektu</b>	<b>2</b>
2.1	Potencjalne grupy użytkowników . . . . .	2
2.2	Wymagania funkcjonalne . . . . .	2
2.3	Wymagania niefunkcjonalne . . . . .	3
2.4	Diagram związków encji . . . . .	3
<b>3</b>	<b>Przykłady realizacji bazy danych</b>	<b>5</b>
3.1	Przykłady zawartości najważniejszych tabel . . . . .	5
3.2	Przykłady kilku zapytań i ich wyników . . . . .	5

# 1 Wprowadzenie

Baza danych przeznaczona jest dla strony internetowej służącej jako platforma do prowadzenia i uczestniczenia w kursach online. Ma ona realizować podstawowe założenia tego typu strony zawierać dane o kursach i ich uczestnikach a także postępie użytkowników w nauce na owych kursach wraz z certyfikatami o ich ukończeniu.

Do realizacji owej bazy posłuży serwer bazodanowy postgresQL za pomocą którego zaimplementuję i przetestuję a także przedstawię sporządzony projekt.



## 2 Opis projektu

Zdalne nauczanie jest obecnie potrzebniejsze niż kiedykolwiek, a więc rośnie też potrzeba na skuteczne narzędzia do prowadzenia zajęć online, stąd pomysł na zaprojektowanie bazy danych właśnie dla tego typu serwisu.

### 2.1 Potencjalne grupy użytkowników

- Administrator – osoba zarządzająca serwisem mająca dostęp do wszystkich danych i pełniąca nadzór nad bazą danych pełni on rolę nadzoru nad zamieszczanymi przez użytkowników treściami
- Twórcy kursów - użytkownicy którzy mają dostęp do tworzenia kursów, a także modułów i innych ich części składowych oraz przypisujący swoje kursy do kategorii
- Prowadzący kursy - użytkownicy którzy nadzorują przebieg kursów mają oni dostęp do danych kursów takich jak zadania czy moduły mogą ustawiać odpowiednie terminy a także sprawdzać i oceniać zadania jak i wystawiać certyfikaty za ukończenie kursu
- Uczestnik kursu - użytkownik który bierze udział w kursie i rozwiązuje w nim zadania ma on dostęp do wyznaczonych mu przez prowadzącego zadań a także do uzyskanych wyników i terminów otwarcia modułów czy też rozpoczęcia zajęć

Warto nadmienić że powyższe role nie wykluczają się wzajemnie administrator może jednocześnie tworzyć i prowadzić kursy, twórca kursu może być jego prowadzącym, a także brać udział w innym kursie.

### 2.2 Wymagania funkcjonalne

Baza danych przede wszystkim przechowuje dane o użytkownikach i o kursach. Do sprawnego funkcjonowania oczywiście potrzebuje również składowych kursów czyli zadań, lekcji i modułów. Ponadto zawiera inne niezbędne dane do funkcjonowania takie jak przynależność użytkowników do kursów czy też kursów do kategorii itp. Pozwala to na stworzenie systemu w którym użytkownicy bez problemu mogą znaleźć odpowiednie kursy rozwiązywać w nich zadania i być nagradzani odpowiednimi certyfikatami.

## 2.3 Wymagania niefunkcjonalne

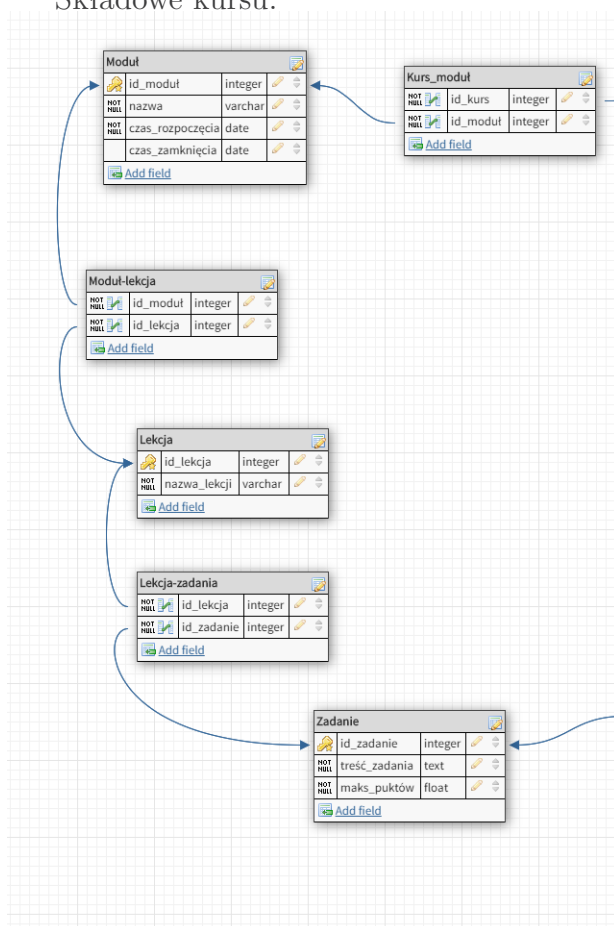
Baza danych została zrealizowana w systemie bazodanowym postgresql 12. Narzędzie to w swoim standardzie zawiera narzędzie PGAdmin jest to interfejs, który pozwala sprawnie zarządzać bazą danych bez użycia wiersza poleceń. Znaczną zaletą tego narzędzia jest prostota w użyciu zawiera dostęp do wprowadzania zapytań i poleceń w języku SQL a także zapewnia dostęp do listy tabel dając wygodny wgląd do bazy danych. Dodatkową zaletą jest fakt iż postgresql obsługuje polskie znaki w nazwach tabel i kolumn co tworzy wygodę we wprowadzaniu danych gdyż nie trzeba owych znaków unikać.



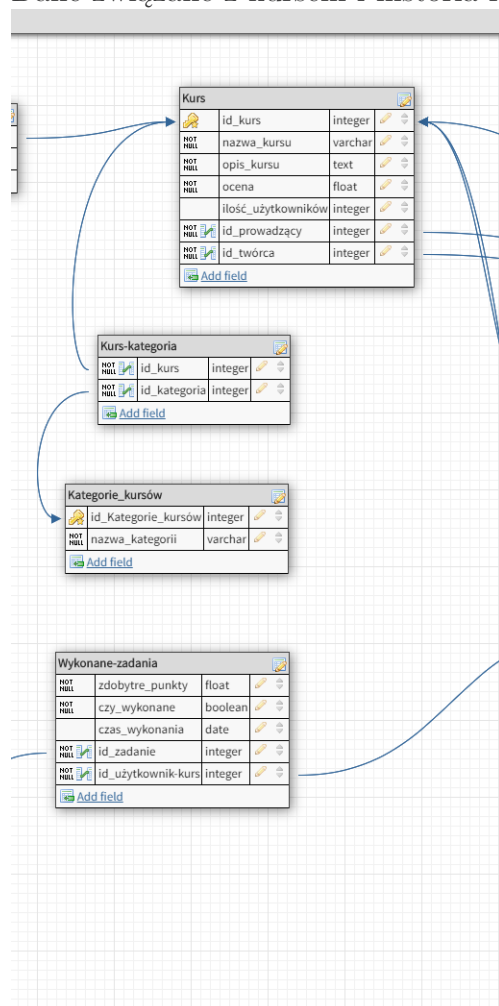
## 2.4 Diagram związków encji

Dla czytelności pełny schemat został podzielony na:

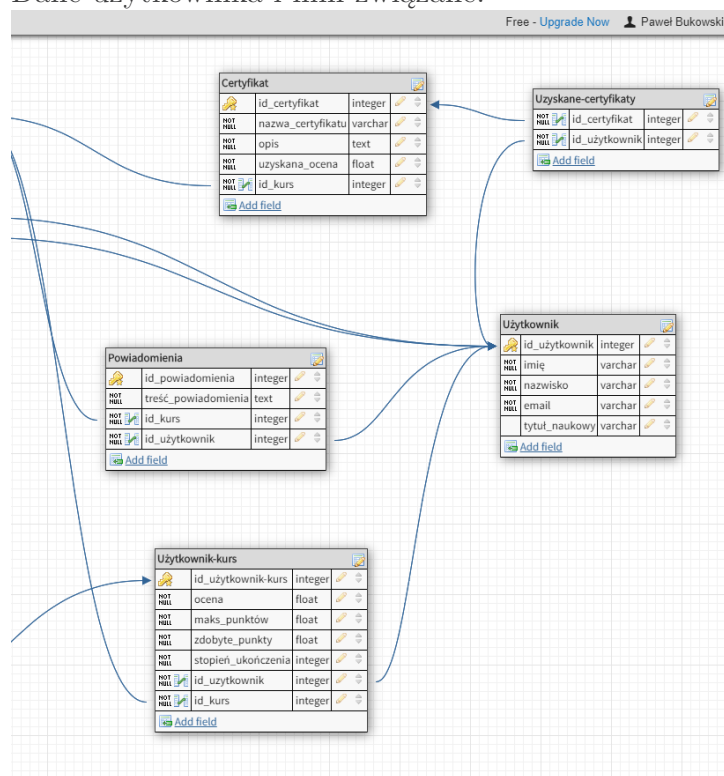
Składowe kursu:



Dane związane z kursem i historia rozwiązań zadań:



Dane użytkownika i nim związane:



## 3 Przykłady realizacji bazy danych

Poniżej znajdują się przykłady pewnych kluczowych tabel i zapytań:

### 3.1 Przykłady zawartości najważniejszych tabel

Tabela z danymi kursów:

Field	Type	Null	Key	Default	Extra
id_kurs	integer	NO	PRI	None	auto_increment
nazwa_kursu	varchar	NO		None	
opis_kursu	text	NO		None	
ocena	float	NO		None	
ilość_użytkowników	integer	NO		None	
id_prowadzący	integer	NO		None	
id_twórca	integer	NO		None	

Tabela z danymi o użytkownikach:

Field	Type	Null	Key	Default	Extra
id_użytkownik	integer	NO	PRI	None	auto_increment
imię	varchar	NO		None	
nazwisko	varchar	NO		None	
email	varchar	NO		None	
tytuł_naukowy	varchar	YES		None	

### 3.2 Przykłady kilku zapytań i ich wyników

Zapytanie o kursy użytkownika o id 3:

```
01 | SELECT nazwa_kursu FROM "Kurs" WHERE id_kurs IN (SELECT id_kurs FROM "
02 |     użytkownik-kurs" WHERE id_użytkownik =3);
03 | -----
04 | Matematyka dyskretna
05 | Algebra liniowa
```

Zapytanie o użytkowników którzy nie stworzyli żadnego kursu:

```
01 | SELECT ęimi, nazwisko FROM "żUytownik" WHERE żid_uytkownik NOT IN (SELECT
    id_twórca FROM "Kurs");
02 | ęimi | nazwisko
03 | -----+-----
04 | Daniel | Marczak
05 | Piotr | Malinowski
06 | Hubert | Ostrowski
07 |
```

Ranking najlepszych kursów/użytkowników(sortowanie po ocenie):

```
01 | SELECT nazwa_kursu FROM "Kurs" ORDER BY ocena DESC;
02 | nazwa_kursu
03 | -----
04 | Bazy danych
05 | Algebra liniowa
06 | Matematyka dyskretna
07 |
```

Lista modułów z kursu bazy danych:

```
01 | SELECT nazwa FROM "łModu" WHERE łid_modu IN (SELECT łid_modu FROM "Kurs-
    łmodu" WHERE id_kurs IN (SELECT id_kurs FROM "Kurs" WHERE nazwa_kursu='
    Bazy danych'));
02 | nazwa
03 | -----
04 | Wyszukiwanie danych w SQL
05 | Tworzenie i edycja tabel w SQL
06 |
```

