

# Academic Data Deliverer

## Bazy Danych

Artur Bednarczyk, Dawid Grajewski, Damian Kwaśniok

Politechnika Śląska

Wydział Matematyki Stosowanej

Informatyka, semestr IV

2 czerwca 2018



**Politechnika  
Śląska**

# Spis treści

<b>1</b>	<b>Opis projektu</b>	<b>3</b>
1.1	Opis . . . . .	3
1.2	Funkcjonalności . . . . .	3
1.2.1	Logowanie i Rejestracja . . . . .	3
1.2.2	Przypasanie do grup . . . . .	3
1.2.3	Notatki . . . . .	3
<b>2</b>	<b>Technologie</b>	<b>3</b>
2.1	Oprogramowanie . . . . .	3
2.2	Technologie . . . . .	3
<b>3</b>	<b>Baza Danych</b>	<b>4</b>
3.1	Diagram Encji . . . . .	4
3.2	Diagram Relacji . . . . .	4
3.3	MySQL . . . . .	5
3.3.1	Zapytania dotyczące 'Colleges' . . . . .	5
3.3.2	Zapytania dotyczące 'Faculties' . . . . .	5
3.3.3	Zapytania dotyczące 'Lecturers' . . . . .	6
3.3.4	Zapytania dotyczące 'Materials' . . . . .	6
3.3.5	Zapytania dotyczące 'Specializations' . . . . .	7
3.3.6	Zapytania dotyczące 'Subjects' . . . . .	7
3.3.7	Zapytania dotyczące 'Users' . . . . .	8
3.3.8	Zapytania dotyczące 'UserSpecializations' . . . . .	8
3.4	Środowisko . . . . .	9
<b>4</b>	<b>Testy</b>	<b>9</b>

# 1 Opis projektu

## 1.1 Opis

Aplikacja dla studentów umożliwiająca szybki i łatwy podgląd dostępnych materiałów! Każdy student może wybrać kierunki studiów, którymi jest zainteresowany i oglądać przypisane do nich notatki! W każdej chwili może dodać nowe lub usunąć nieinteresujące go kierunki ze swojej listy. Aplikacja jest prosta w użytkowaniu, jednak wymaga połączenia z internetem.

## 1.2 Funkcjonalności

### 1.2.1 Logowanie i Rejestracja

Załącz swoje konto, a będziesz mógł korzystać z aplikacji z każdego urządzenia, na którym jest zainstalowana i posiada dostęp do internetu! Zachowaj Twoje listy kierunków na swoim koncie!

### 1.2.2 Przypasanie do grup

Wybierz kierunki, które Cię interesują i przeglądaj materiały z nimi powiązane. W każdej chwili możesz dopisać do swojej listy nowe kierunki lub usunąć już niepotrzebne.

### 1.2.3 Notatki

Przeglądaj dostępne materiały z listy kierunków, którą sam utworzyłeś. Jeśli chcesz mieć dostęp offline to pobierz materiał w formie pliku!

# 2 Technologie

## 2.1 Oprogramowanie

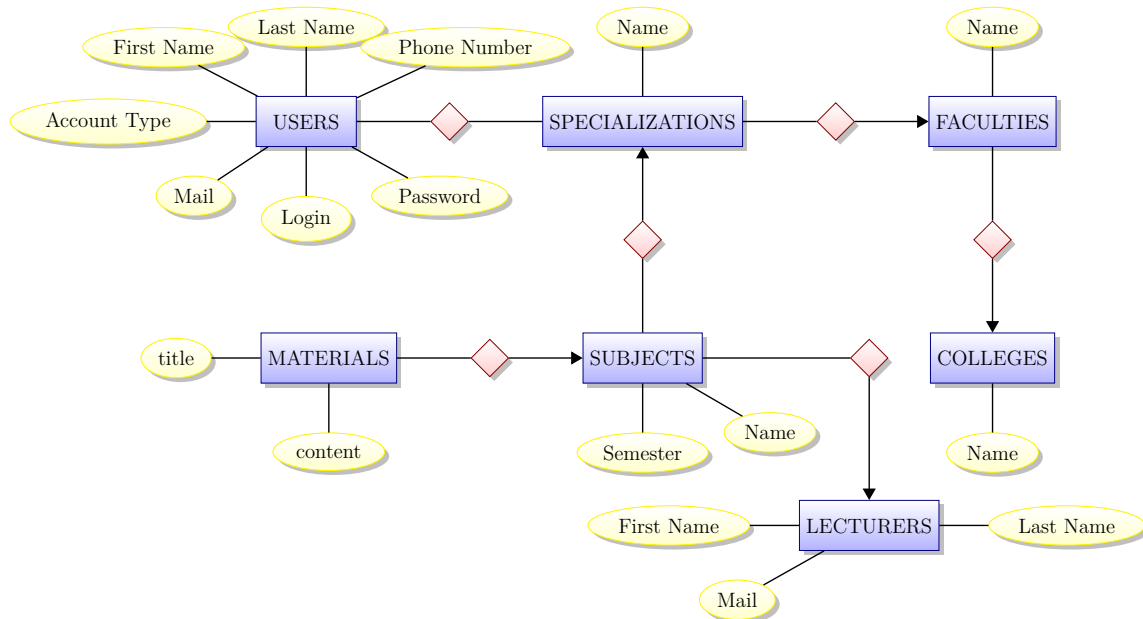
- Visual Studio 2015 - Środowisko programistyczne.
- SourceTree - Kontrola wersji.
- GitHub - Repozytorium do przechowywania wersji online.
- Heroku - Chmura, w której przechowywana jest baza danych.

## 2.2 Technologie

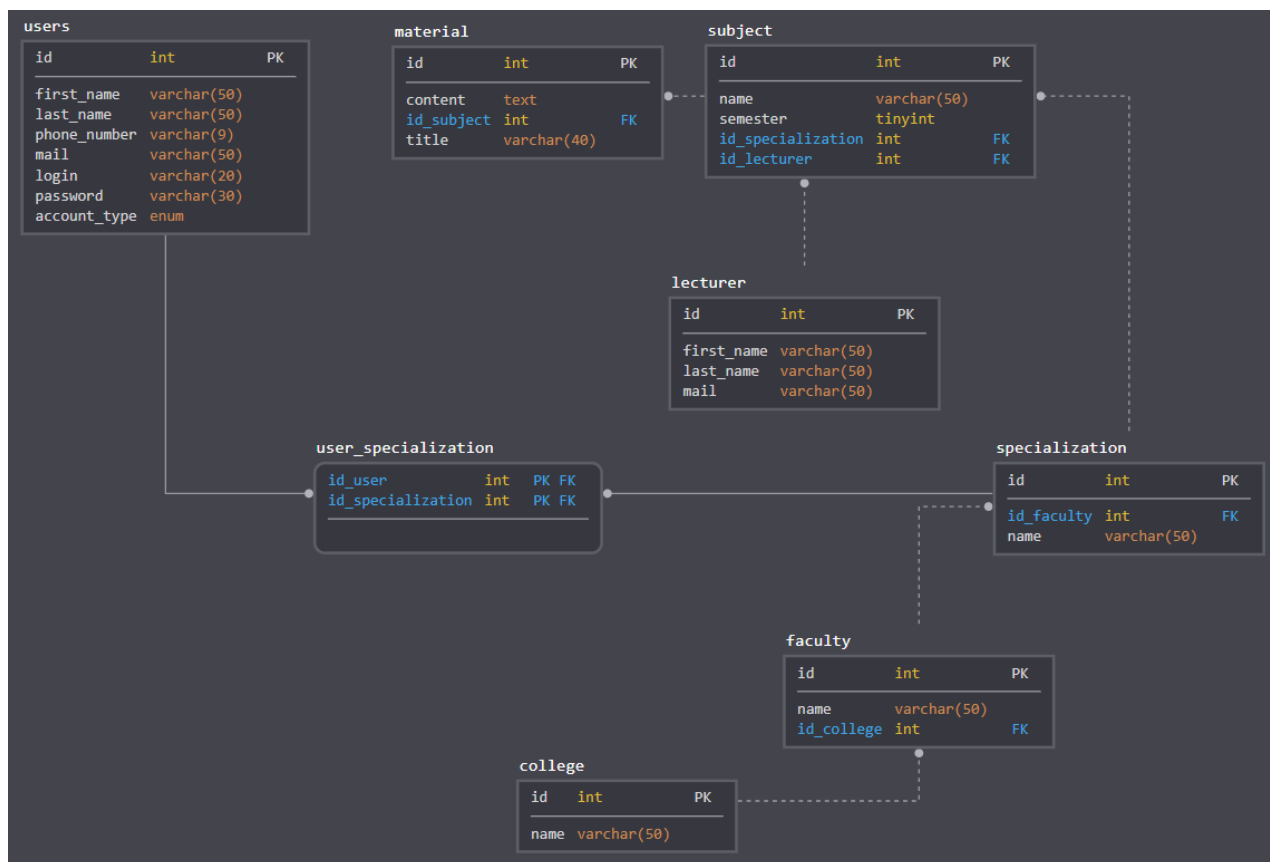
- C#
- .NET
- MySQL

## 3 Baza Danych

### 3.1 Diagram Encji



### 3.2 Diagram Relacji



## 3.3 MySQL

### 3.3.1 Zapytania dotyczące 'Colleges'

---

```
SP_GET_ALL = "SELECT 'Id', 'Name' FROM colleges";
SP_GET_FILTER = "SELECT 'Id', 'Name' FROM colleges {0}";
SP_GET_BYPAGE = "SELECT * FROM colleges {1} ORDER BY {0} LIMIT {2},
{3}; SELECT COUNT(*) FROM colleges {1}";
SP_GET_BYID = "SELECT 'Id', 'Name' FROM colleges WHERE Id =
@ref_id";
SP_ADD = "INSERT INTO colleges ( Id, Name) VALUES ( @Id, @Name) ";
SP_ADD1 = "INSERT INTO colleges ( Id, Name) VALUES ( @Id, @Name)
SELECT @Id = @@IDENTITY";
SP_UPDATE = "UPDATE colleges SET Name = @Name WHERE Id = @Id";
SP_DELETE = "DELETE FROM colleges WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM colleges {0}";
SP_GET_LOOKUP = "SELECT Id, Name FROM colleges";
```

---

### 3.3.2 Zapytania dotyczące 'Faculties'

---

```
SP_GET_ALL = "SELECT 'Id', 'Name', 'College_Id' FROM faculties";
SP_GET_FILTER = "SELECT 'Id', 'Name', 'College_Id' FROM faculties
{0}";
SP_GET_BYPAGE = "SELECT * FROM faculties {1} ORDER BY {0} LIMIT
{2}, {3}; SELECT COUNT(*) FROM faculties {1}";
SP_GET_BYID = "SELECT 'Id', 'Name', 'College_Id' FROM faculties
WHERE Id = @ref_id";
SP_ADD = "INSERT INTO faculties ( Id, Name, College_Id) VALUES (
@Id, @Name, @College_Id) ";
SP_ADD1 = "INSERT INTO faculties ( Id, Name, College_Id) VALUES (
@Id, @Name, @College_Id) SELECT @Id = @@IDENTITY";
SP_UPDATE = "UPDATE faculties SET Name = @Name, College_Id =
@College_Id WHERE Id = @Id";
SP_DELETE = "DELETE FROM faculties WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM faculties {0}";
SP_GET_LOOKUP = "SELECT Id, Name FROM faculties";
```

---

### 3.3.3 Zapytania dotyczące 'Lecturers'

---

```
SP_GET_ALL = "SELECT  'Id', 'Name', 'Surname', 'EmailAddress' FROM
lecturers";
SP_GET_FILTER = "SELECT  'Id', 'Name', 'Surname', 'EmailAddress'
FROM lecturers {0}";
SP_GET_BYPAGE = "SELECT * FROM lecturers {1} ORDER BY {0} LIMIT
{2}, {3}; SELECT COUNT(*) FROM lecturers {1}";
SP_GET_BYID = "SELECT  'Id', 'Name', 'Surname', 'EmailAddress' FROM
lecturers WHERE Id = @ref_id";
SP_ADD = "INSERT INTO lecturers ( Id, Name, Surname, EmailAddress)
VALUES ( @Id, @Name, @Surname, @EmailAddress) ";
SP_ADD1 = "INSERT INTO lecturers ( Id, Name, Surname, EmailAddress)
VALUES ( @Id, @Name, @Surname, @EmailAddress) SELECT @Id =
@@IDENTITY";
SP_UPDATE = "UPDATE lecturers SET Name = @Name, Surname = @Surname,
EmailAddress = @EmailAddress WHERE Id = @Id";
SP_DELETE = "DELETE FROM lecturers WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM lecturers {0}";
SP_GET_LOOKUP = "SELECT Id, Name FROM lecturers";
```

---

### 3.3.4 Zapytania dotyczące 'Materials'

---

```
SP_GET_ALL = "SELECT  'Id', 'Title', 'Content', 'Subject_Id' FROM
materials";
SP_GET_FILTER = "SELECT  'Id', 'Title', 'Content', 'Subject_Id'
FROM materials {0}";
SP_GET_BYPAGE = "SELECT * FROM materials {1} ORDER BY {0} LIMIT
{2}, {3}; SELECT COUNT(*) FROM materials {1}";
SP_GET_BYID = "SELECT  'Id', 'Title', 'Content', 'Subject_Id' FROM
materials WHERE Id = @ref_id";
SP_ADD = "INSERT INTO materials ( Id, Title, Content, Subject_Id)
VALUES ( @Id, @Content, @Subject_Id) ";
SP_ADD1 = "INSERT INTO materials ( Id, Title, Content, Subject_Id)
VALUES ( @Id, @Content, @Subject_Id) SELECT @Id = @@IDENTITY";
SP_UPDATE = "UPDATE materials SET Title = @Title, Content =
@Content, Subject_Id = @Subject_Id WHERE Id = @Id";
SP_DELETE = "DELETE FROM materials WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM materials {0}";
SP_GET_LOOKUP = "SELECT Id, Title, Content FROM materials";
```

---

### 3.3.5 Zapytania dotyczące 'Specializations'

---

```
SP_GET_ALL = "SELECT  'Id', 'Name', 'Faculty_Id' FROM
    specializations";
SP_GET_FILTER = "SELECT  'Id', 'Name', 'Faculty_Id' FROM
    specializations {0}";
SP_GET_BYPAGE = "SELECT * FROM specializations {1} ORDER BY {0}
    LIMIT {2}, {3}; SELECT COUNT(*) FROM specializations {1}";
SP_GET_BYID = "SELECT  'Id', 'Name', 'Faculty_Id' FROM
    specializations WHERE Id = @ref_id";
SP_ADD = "INSERT INTO specializations ( Id, Name, Faculty_Id)
    VALUES ( @Id, @Name, @Faculty_Id) ";
SP_ADD1 = "INSERT INTO specializations ( Id, Name, Faculty_Id)
    VALUES ( @Id, @Name, @Faculty_Id) SELECT @Id = @@IDENTITY";
SP_UPDATE = "UPDATE specializations SET Name = @Name, Faculty_Id =
    @Faculty_Id WHERE Id = @Id";
SP_DELETE = "DELETE FROM specializations WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM specializations {0}";
SP_GET_LOOKUP = "SELECT Id, Name FROM specializations";
```

---

### 3.3.6 Zapytania dotyczące 'Subjects'

---

```
SP_GET_ALL = "SELECT  'Id', 'Name', 'Semester', 'Lecturer_Id',
    'Specialization_Id' FROM subjects";
SP_GET_FILTER = "SELECT  'Id', 'Name', 'Semester', 'Lecturer_Id',
    'Specialization_Id' FROM subjects {0}";
SP_GET_BYPAGE = "SELECT * FROM subjects {1} ORDER BY {0} LIMIT {2},
    {3}; SELECT COUNT(*) FROM subjects {1}";
SP_GET_BYID = "SELECT  'Id', 'Name', 'Semester', 'Lecturer_Id',
    'Specialization_Id' FROM subjects WHERE Id = @ref_id";
SP_ADD = "INSERT INTO subjects ( Id, Name, Semester, Lecturer_Id,
    Specialization_Id) VALUES ( @Id, @Name, @Semester, @Lecturer_Id,
    @Specialization_Id) ";
SP_ADD1 = "INSERT INTO subjects ( Id, Name, Semester, Lecturer_Id,
    Specialization_Id) VALUES ( @Id, @Name, @Semester, @Lecturer_Id,
    @Specialization_Id) SELECT @Id = @@IDENTITY";
SP_UPDATE = "UPDATE subjects SET Name = @Name, Semester =
    @Semester, Lecturer_Id = @Lecturer_Id, Specialization_Id =
    @Specialization_Id WHERE Id = @Id";
SP_DELETE = "DELETE FROM subjects WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM subjects {0}";
SP_GET_LOOKUP = "SELECT Id, Name FROM subjects";
```

---

### 3.3.7 Zapytania dotyczące 'Users'

---

```
SP_GET_ALL = "SELECT 'Id', 'FirstName', 'LastName', 'PhoneNumber',
'EmailAddress', 'Login', 'Password', 'AccountType' FROM users";
SP_GET_FILTER = "SELECT 'Id', 'FirstName', 'LastName',
'PhoneNumber', 'EmailAddress', 'Login', 'Password', 'AccountType'
FROM users {0}";
SP_GET_BYPAGE = "SELECT * FROM users {1} ORDER BY {0} LIMIT {2},
{3}; SELECT COUNT(*) FROM users {1}";
SP_GET_BYID = "SELECT 'Id', 'FirstName', 'LastName',
'PhoneNumber', 'EmailAddress', 'Login', 'Password', 'AccountType'
FROM users WHERE Id = @ref_id";
SP_ADD = "INSERT INTO users ( Id, FirstName, LastName, PhoneNumber,
EmailAddress, Login, Password, AccountType) VALUES ( @Id,
@FirstName, @LastName, @PhoneNumber, @EmailAddress, @Login,
@Password, @AccountType) ";
SP_ADD1 = "INSERT INTO users ( Id, FirstName, LastName,
PhoneNumber, MailAddress, Login, Password, AccountType) VALUES (
@Id, @FirstName, @LastName, @PhoneNumber, @MailAddress, @Login,
@Password, @AccountType) SELECT @Id = @@IDENTITY";
SP_UPDATE = "UPDATE users SET FirstName = @FirstName, LastName =
@LastName, PhoneNumber = @PhoneNumber, MailAddress =
@MailAddress, Login = @Login, Password = @Password, AccountType =
@AccountType WHERE Id = @Id";
SP_DELETE = "DELETE FROM users WHERE Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM users {0}";
SP_GET_LOOKUP = "SELECT Id, FirstName FROM users";
```

---

### 3.3.8 Zapytania dotyczące 'UserSpecializations'

---

```
SP_GET_ALL = "SELECT 'User_Id', 'Specialization_Id' FROM
userspecializations";
SP_GET_FILTER = "SELECT 'User_Id', 'Specialization_Id' FROM
userspecializations {0}";
SP_GET_BYPAGE = "SELECT * FROM userspecializations {1} ORDER BY {0}
LIMIT {2}, {3}; SELECT COUNT(*) FROM userspecializations {1}";
SP_GET_BYID = "SELECT 'User_Id', 'Specialization_Id' FROM
userspecializations WHERE User_Id = @ref_id";
SP_ADD = "INSERT INTO userspecializations ( User_Id,
Specialization_Id) VALUES ( @User_Id, @Specialization_Id) ";
SP_ADD1 = "INSERT INTO userspecializations ( User_Id,
Specialization_Id) VALUES ( @User_Id, @Specialization_Id) SELECT
@User_Id = @@IDENTITY";
SP_UPDATE = "UPDATE userspecializations SET Specialization_Id =
@Specialization_Id WHERE User_Id = @User_Id";
SP_DELETE = "DELETE FROM userspecializations WHERE User_Id=@ref_id";
SP_DELETE_FILTER = "DELETE FROM userspecializations {0}";
SP_GET_LOOKUP = "SELECT User_Id, User_Id FROM userspecializations";
```

---



---

### 3.4 Środowisko

Baza danych znajduje się w chmurze Heroku. Łączymy się z nią za pomocą...

## 4 Testy

Baza działa... Jakies info z Heroku można tutaj dać.