

## Temat projektu

Implementacja zarządzania uczelnią.

## Użyte narzędzia

- Java (IntelliJ IDEA)
- SQL Developer
- Oracle DB
- GitHub - <https://github.com/lutyjj/BD1>

## Zakres projektu

Zadaniem projektu jest implementacja bazy danych oraz tworzenie aplikacji, pozwalającej na zarządzanie uczelnią.

## Wymagania funkcjonalne projektowanego systemu

Została stworzona baza danych reprezentująca działanie uczelni ze strony administracyjnej. W danym systemie jako aktor występuje administracja uczelni, która ma pełny dostęp do bazy danych i może ją edytować za pomocą aplikacji.

Aplikacja pozwala na dodawanie studenta, skreślenie z listy studentów; tworzenie, zamykanie i napełnienie grup studentami.

Dodatkową funkcją aplikacji jest wygenerowanie rozkładu zajęć dla studenta i prowadzącego. Aplikacja analizuje wszystkie grupy i sprawdza czy znajduje się student albo prowadzący należy do tej albo innej grupy. Na podstawie otrzymanych danych jest tworzony rozkład zajęć.

## Szczegółowy opis ról

Pod tabelami są umieszczone nie oczywiste pola zawarte w tabelach.

### Students

STUDENTS						
COLUMN_NAME	STUDENT_ID	FIRST_NAME	LAST_NAME	FACULTY_ID	FIELD_NAME	STARTING_DATE
DATA_TYPE	NUMBER (6, 0)	VARCHAR2	VARCHAR2	CHAR	VARCHAR2	DATE
NULLABLE	NO	NO	NO	YES	YES	NO

**FIELD\_NAME** – kierunek studiów.

**STARTING\_DATE** – data rozpoczęcia studiów. Może być NULL, znaczy student już ukończył studia lub został skreślony z listy studentów.

### Professors

PROFESSORS					
COLUMN_NAME	PROFESSOR_ID	FIRST_NAME	LAST_NAME	FACULTY_ID	DEPARTMENT_NAME

DATA_TYPE	NUMBER (6, 0)	VARCHAR2	VARCHAR2	CHAR	VARCHAR2
NULLABLE	NO	NO	NO	YES	YES

**DEPARTMENT\_NAME** – nazwa katedry, do której należy prowadzący.

### Subjects

SUBJECTS				
COLUMN_NAME	SUBJECT_ID	SUBJECT_NAME	FACULTY_ID	FIELD_NAME
DATA_TYPE	NUMBER (6, 0)	VARCHAR2	CHAR	VARCHAR2
NULLABLE	NO	NO	YES	YES

**FIELD\_NAME** – kierunek studiów, do którego należy przedmiot.

### Groups

GROUPS				
COLUMN_NAME	GROUP_ID	SUBJECT_ID	PROFESSOR_ID	PARITY
DATA_TYPE	NUMBER (6, 0)	NUMBER (6, 0)	NUMBER (6, 0)	VARCHAR2
NULLABLE	NO	YES	YES	YES

GROUPS				
COLUMN_NAME	DAY	TIME	SUBJECT_FORM	STUDENT_LIMIT
DATA_TYPE	NUMBER	NUMBER	VARCHAR2	NUMBER (6, 0)
NULLABLE	YES	YES	NO	YES

**PARITY** – parzystość tygodnia.

**SUBJECT\_FORM** – forma zajęć (wykład, laboratorium, seminarium, ćwiczenia).

**STUDENT\_LIMIT** – maksymalna ilość studentów. Może być NULL, gdy liczba jest nieograniczona (wykłady ogólnounuczelniane) lub gdy grupa nie jest jeszcze otwarta.

### Faculties

FACULTIES			
COLUMN_NAME	FACULTY_ID	FACULTY_NAME	DEAN_ID
DATA_TYPE	CHAR	VARCHAR2	NUMBER(6, 0)
NULLABLE	NO	NO	YES

**FACULTY\_DEAN** – dziekan wydziału.

### Students log

STUDENTS_LOG				
COLUMN_NAME	STUDENT_ID	STARTING_DATE	ENDING_DATE	FACULTY_ID
DATA_TYPE	NUMBER (6, 0)	DATE	DATE	CHAR
NULLABLE	NO	NO	NO	YES

W logu zapisujemy studentów, którzy zostali skreślone z listy studentów lub ukończyli studia. ENDING\_DATE służy dla obu przypadków.

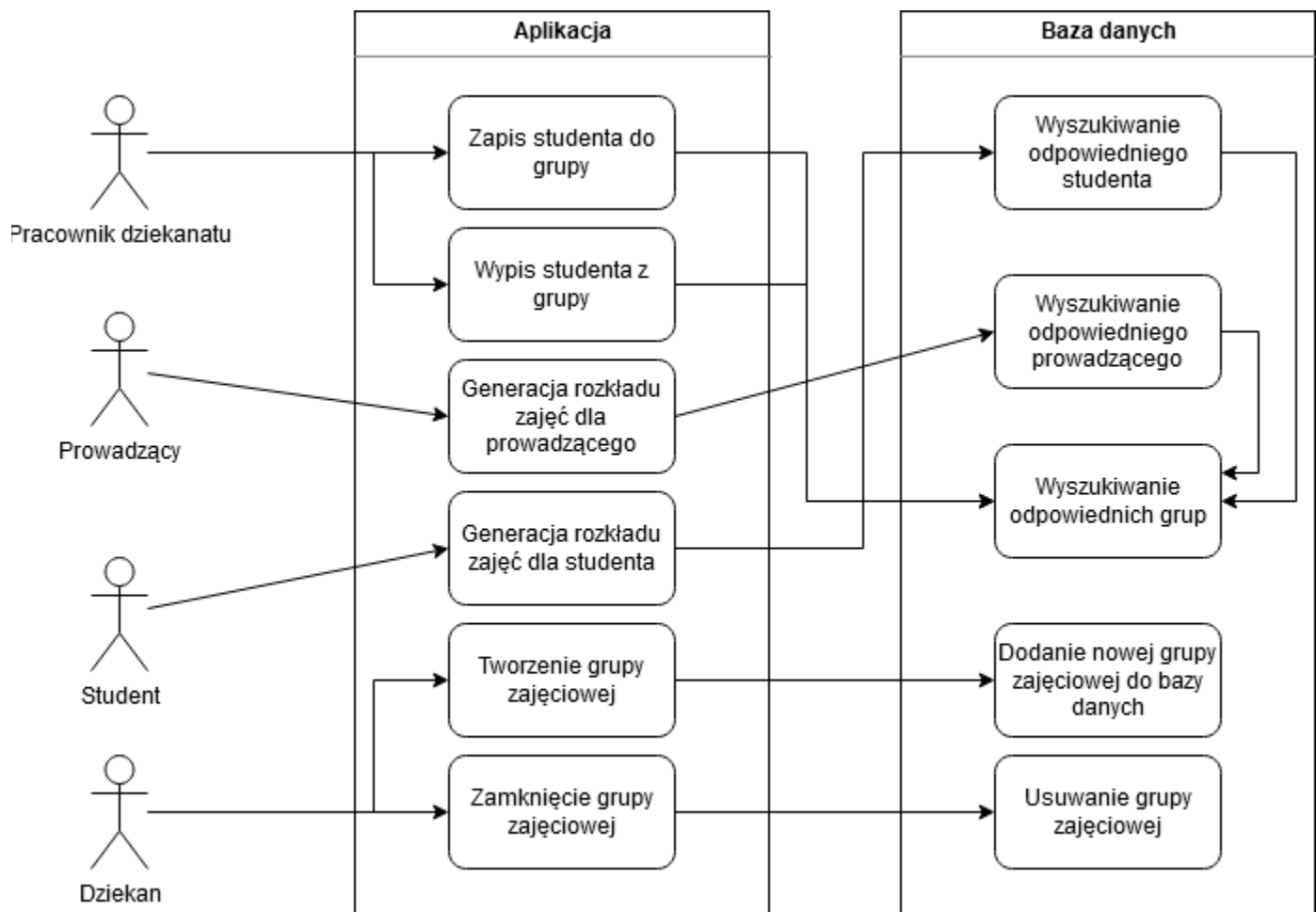
Do tej tabeli używane są triggery, które ze zmianą FACULTY\_ID na NULL zapisują dane studenta.

### Records

RECORDS			
COLUMN_NAME	RECORD_DATE	GROUP_ID	STUDENT_ID
DATA_TYPE	CHAR	NUMBER(6, 0)	NUMBER(6, 0)
NULLABLE	NO	NO	NO

Records służy do przypisania studentów do odpowiednich grup.

### Diagram przypadków użycia



## Określenie metodyki oraz notacji

Użyta notacja ERD - Bachman.

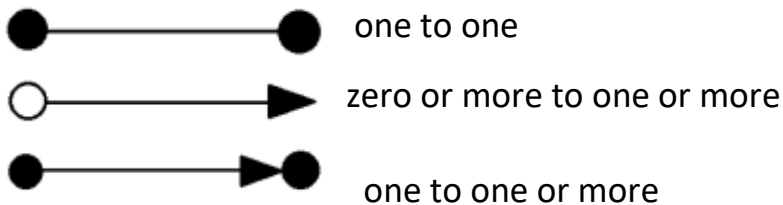
### Bloki logiczne

Podmioty - reprezentacja klasy rzeczywistych lub abstrakcyjnych rzeczy (ludzi, przedmiotów, miejsc, wydarzeń, idei, kombinacja rzeczy, etc.), które są rozpoznawane jako instancje tej samej klasy, ponieważ mają takie same właściwości i mogą uczestniczyć w tych samych relacjach.

Używaliśmy podmiotów niezależnych od identyfikatorów.

### Relacje

Relaje w notacji Bachman są wyrażone w ilości powiązań pomiędzy cegiełkami:



Tablica **STUDENTS** jest związana z tablicą **STUDENTS\_LOG** za pomocą klucza podstawowego **STUDENT\_ID** z tablicą **STUDENTS\_LOG** i **RECORDS** która służy do notacji powiązań student-grupa.

Tablica **SUBJECTS** jest związana z tablicą **GROUPS** za pomocą klucza podstawowego **GROUPS\_ID**.

Tablica **FACULTIES** jest związana z tablicami **STUDENTS**, **PROFESSORS**, **STUDENTS\_LOG**, **SUBJECTS** za pomocą klucza podstawowego **FACULTY\_ID**.

Tablica **PROFESSORS** jest związana z tablicą **GROUPS** za pomocą klucza podstawowego **PROFESOR\_ID**.

## **Legenda:**

*Faculties* → *Students*: Studiuje na wydziale → Jest studentem wydziału

*Faculties* → *Professors*: Dziekan wydziału -> Pracuje na wydziale

*Professors* → *Faculties*: Pracuje na wydziale → Pradocawca

*Professors* → *Groups*: Pracownik prowadzi przedmiot w → Grupa, w której prowadzi

*Subjects* → *Groups*: Przedmiot prowadzony w → Grupa, w której będą przedmiot uczyć

*Faculties* → *Subject*: Wydział na którym prowadzą zajęcia → Zajęcia, prowadzone na wydziale

*Groups* → *Records*: W grupie zajęciowej studiują → Związek studenta z grupą

*Students* → *Records*: Student zapisany do → Związek grupy z studentem

*Students* → *Students\_Log*: Studiuje z i do → Był studentem

*Faculties* → *Student\_Log*: Studiuje na wydziale → Był na wydziale