

**UNIwersYTET RZESZOWSKI**  
**WYDZIAŁ NAUK ŚCISŁYCH I TECHNICZNYCH**  
**INSTYTUT INFORMATYKI**



*Mykhailo Kleban*  
134922

*Informatyka*

*System rezerwacji sal/podział godzin*

Praca projektowa

Praca wykonana pod kierunkiem  
dr inż. Ewa Żesławska

Rzeszów 2025



# Spis treści

<b>1. Wstęp</b>	2
1.1. Cel projektu	2
1.2. Zakres funkcjonalny systemu	2
1.3. Zastosowane technologie	2
1.4. Struktura dokumentacji	2
<b>2. Opis projektu</b>	3
2.1. Cel i przeznaczenie	3
2.2. Główne funkcjonalności	3
2.3. Typy zajęć	3
2.4. Zastosowane technologie	3
<b>3. Implementacja systemu</b>	4
3.1. Struktura aplikacji	4
<b>4. Opis interfejsu użytkownika</b>	5
4.1. Baza danych	11
4.2. Diagram Baza Danych	11
4.3. Interfejs użytkownika	12
4.4. Walidacja i błędy	12
4.5. Szczegółowy opis komponentów GUI	12
4.5.1. Formularz Dodawania Zajęć (DodajZajeciaPanel)	12
4.5.2. Formularz Edytowania Zajęć (EdytujZajeciaPanel)	12
4.5.3. Formularz Logowania (Login)	13
4.5.4. Panel Sekretariatu (SekretariatPanel)	13
4.6. Diagram komponentów GUI i zależności	13
<b>Bibliografia</b>	14
<b>Spis rysunków</b>	14
<b>Spis tabel</b>	15
<b>Spis listingów</b>	16
<b>Oświadczenie studenta o samodzielności pracy</b>	17

# 1. Wstęp

## 1.1. Cel projektu

Celem projektu było zaprojektowanie i zaimplementowanie aplikacji desktopowej wspomagającej zarządzanie zajęciami akademickimi oraz rezerwację sal dydaktycznych w środowisku uczelni wyższej. System ma za zadanie ułatwić pracownikom sekretariatu organizację planów zajęć, eliminując problemy związane z ręcznym układaniem grafiku.

## 1.2. Zakres funkcjonalny systemu

Dzięki zastosowanym funkcjonalnościom możliwe jest m.in.:

- dodawanie nowych zajęć do bazy danych,
- filtrowanie według różnych kryteriów (typ, sala, grupa),
- edytowanie istniejących rekordów,
- sprawdzanie dostępności sal i konfliktów czasowych.

Aplikacja automatycznie weryfikuje poprawność danych wejściowych oraz wyświetla komunikaty błędów w przypadku wykrycia kolizji.

## 1.3. Zastosowane technologie

Projekt został zrealizowany w języku **Java**, przy użyciu biblioteki **Swing** do stworzenia graficznego interfejsu użytkownika. Do komunikacji z bazą danych zastosowano technologię **JDBC**, natomiast dane przechowywane są w relacyjnej bazie danych **PostgreSQL**.

## 1.4. Struktura dokumentacji

Dokumentacja została przygotowana z użyciem systemu składu tekstu  $\text{\LaTeX}$  i zawiera:

- opis struktury aplikacji,
- diagramy klas i bazy danych,
- opis interfejsu użytkownika,
- opis komponentów i funkcjonalności systemu.

## 2. Opis projektu

### 2.1. Cel i przeznaczenie

Projekt **System rezerwacji sal / podział godzin** został zaprojektowany z myślą o ułatwieniu zarządzania harmonogramem zajęć akademickich. Głównym celem systemu jest wsparcie administracji uczelni w planowaniu i koordynowaniu zajęć dydaktycznych w sposób zautomatyzowany i intuicyjny.

### 2.2. Główne funkcjonalności

System umożliwia użytkownikowi wykonywanie następujących operacji:

- dodawanie zajęć wraz z informacjami: dzień tygodnia, godzina, typ zajęć, kierunek, przedmiot, prowadzący, sala, grupa;
- edytowanie oraz usuwanie wcześniej dodanych zajęć;
- filtrowanie zajęć według sali, grupy i typu zajęć;
- walidację danych przy wprowadzaniu (np. sprawdzanie konfliktów sal i grup);
- obsługę wyjątków i prezentację komunikatów błędów.

### 2.3. Typy zajęć

System rozróżnia trzy typy zajęć, które różnią się liczbą przypisanych grup:

- **Wykład (Wykład)** – przeznaczony dla wszystkich grup;
- **Projekt (ćwiczenia)** – przeznaczony dla dwóch konkretnych grup (np. A i B);
- **Laboratorium** – przeznaczone dla jednej grupy.

### 2.4. Zastosowane technologie

Do realizacji projektu wykorzystano następujące technologie:

- język **Java**;
- biblioteka **Swing** do budowy graficznego interfejsu użytkownika;
- baza danych **PostgreSQL**;
- interfejs **JDBC** do komunikacji z bazą danych.

## 3. Implementacja systemu

System został zaimplementowany w języku **Java**, z wykorzystaniem biblioteki **Swing** do budowy graficznego interfejsu użytkownika oraz technologii **JDBC** do komunikacji z bazą danych **PostgreSQL**.

### 3.1. Struktura aplikacji

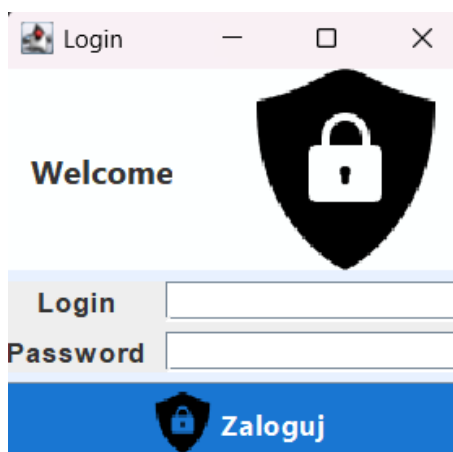
Struktura projektu została logicznie podzielona na pakiety zgodnie z zasadami dobrej organizacji kodu. W folderze `src` znajdują się wszystkie elementy źródłowe aplikacji, zorganizowane w następujący sposób:

- `dao` – klasy odpowiedzialne za dostęp do danych:
  - `LoginDAO` – obsługa uwierzytelniania użytkownika;
  - `ZajeciaDAO` – operacje CRUD na tabeli zajęć.
- `database` – logika połączenia z bazą danych:
  - `DatabaseConnection` – klasa łącząca aplikację z PostgreSQL.
- `DodajZajeciaPanel` – komponent GUI odpowiedzialny za dodawanie nowych zajęć:
  - `DodajZajeciaPanel.java/.form` – panel formularza oraz jego widok.
- `EdytujZajeciaPanel` – komponent GUI służący do edycji zajęć:
  - `EdytujZajeciaPanel.java/.form` – logika i widok edycji.
- `Login` – komponent odpowiedzialny za ekran logowania:
  - `Login.java/.form` – widok i obsługa logowania.
- `model` – klasy reprezentujące dane biznesowe:
  - `Zajecia` – klasa bazowa reprezentująca ogólne zajęcia;
  - `Wyklad`, `Projekt`, `Laboratorium` – klasy dziedziczące;
  - `PlanZajec` – klasa pomocnicza reprezentująca pojedynczy wpis w planie.
- `SekretariatPanel` – główny interfejs do zarządzania zajęciami:
  - `SekretariatPanel.java/.form` – widok panelu oraz jego logika.
- `resource` – folder przechowujący zasoby zewnętrzne (np. ikony lub grafiki).
- `Main.java` – klasa uruchamiająca aplikację.

Takie rozdzielenie pozwala na lepszą czytelność kodu, łatwiejsze zarządzanie komponentami oraz zgodność z zasadami programowania obiektowego.

## 4. Opis interfejsu użytkownika

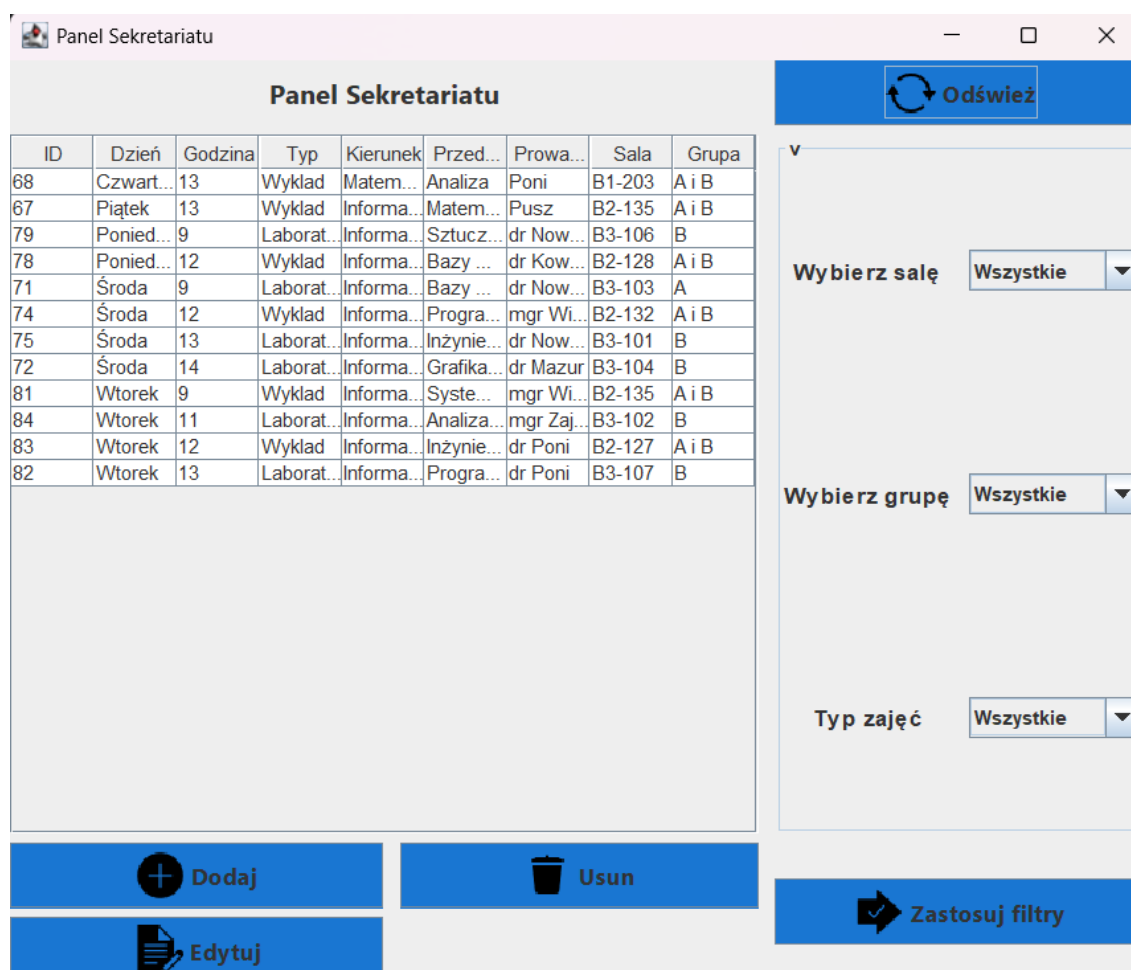
### Panel logowania



Rys. 4.1. Panel logowania do systemu

Panel logowania umożliwia autoryzację użytkownika przed uzyskaniem dostępu do głównego interfejsu systemu. Użytkownik wprowadza dane w pola **Login** i **Password**, a następnie klika przycisk **Zaloguj**. Przy nieprawidłowych danych wyświetlany jest komunikat błędu.

## Główny panel sekretariatu



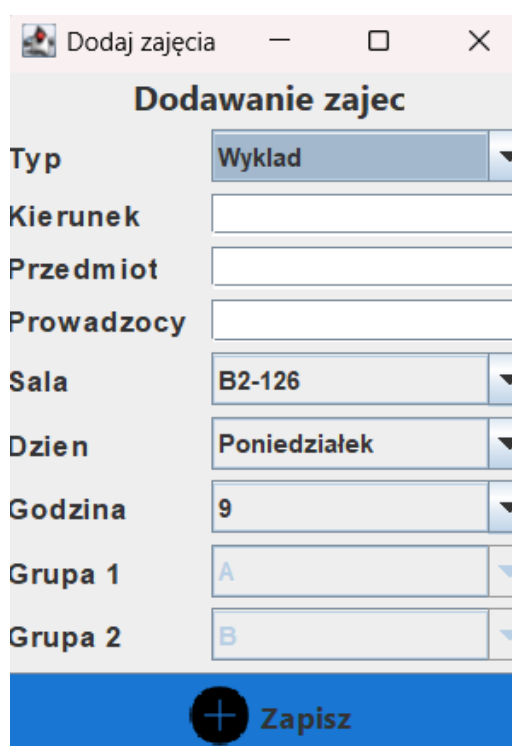
Rys. 4.2. Główny panel sekretariatu z listą zajęć i filtrowaniem

Główny panel aplikacji wyświetla listę wszystkich zajęć w formie tabeli. Użytkownik może:

- **Dodawać zajęcia** – klikając przycisk **Dodaj**,
- **Edytować istniejące zajęcia** – po zaznaczeniu wiersza i kliknięciu **Edytuj**,
- **Usuwać zajęcia** – klikając przycisk **Usuń**,
- **Filtrować dane** – za pomocą rozwijanych list: sala, grupa, typ zajęć,
- **Odświeżać widok** – klikając przycisk **Odśwież**.



## Formularz dodawania zajęć typu Wykład



**Dodawanie zajęć**

Typ: Wykład

Kierunek:

Przedmiot:

Prowadzocy:

Sala: B2-126

Dzień: Poniedziałek

Godzina: 9

Grupa 1: A

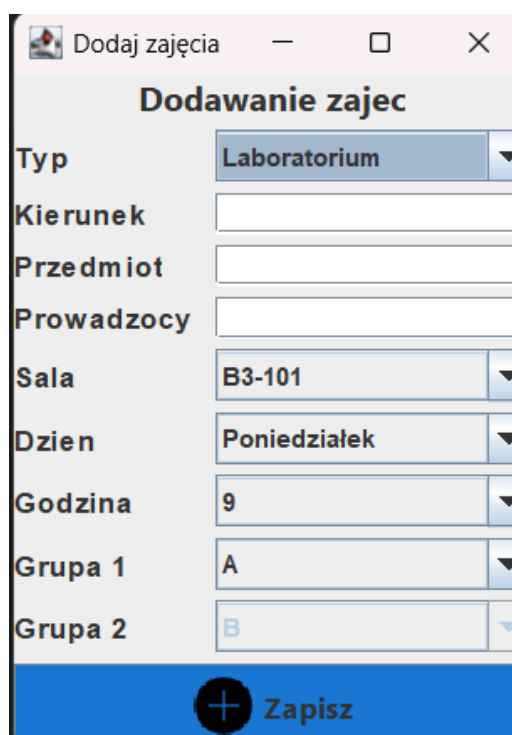
Grupa 2: B

**+ Zapisz**

Rys. 4.3. Formularz dodawania zajęć typu Wykład

Użytkownik wprowadza dane dotyczące wykładu: kierunek, przedmiot, prowadzący, sala, dzień tygodnia, godzina. Zajęcia typu wykład są przypisane do wszystkich grup, więc pola Grupa 1 i Grupa 2 są wyszarzone.

## Formularz dodawania zajęć typu Laboratorium

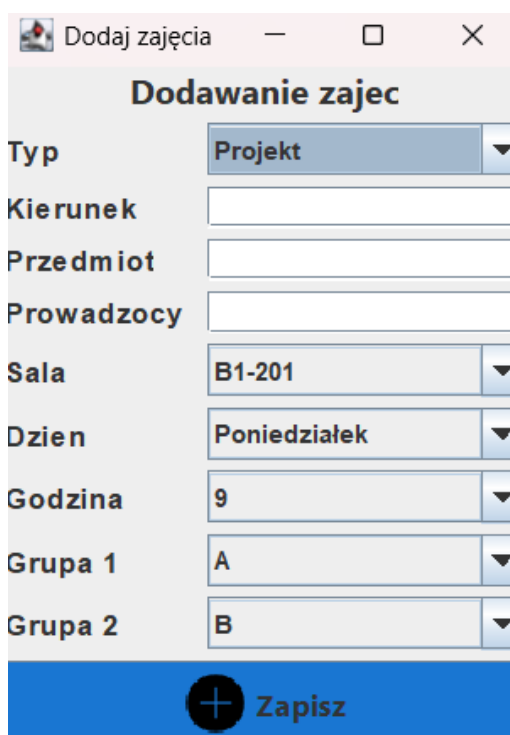


The image shows a software window titled "Dodaj zajęcia" (Add classes). Inside, the title "Dodawanie zajec" (Adding classes) is displayed. The form contains several fields: "Typ" (Type) is a dropdown menu set to "Laboratorium"; "Kierunek" (Direction) is an empty text field; "Przedmiot" (Subject) is an empty text field; "Prowadzocy" (Instructors) is an empty text field; "Sala" (Room) is a dropdown menu set to "B3-101"; "Dzien" (Day) is a dropdown menu set to "Poniedziałek" (Monday); "Godzina" (Hour) is a dropdown menu set to "9"; "Grupa 1" (Group 1) is a dropdown menu set to "A"; and "Grupa 2" (Group 2) is a dropdown menu set to "B". At the bottom of the form is a large blue button with a white plus icon and the text "Zapisz" (Save).

Rys. 4.4. Formularz dodawania zajęć typu Laboratorium

W tym formularzu użytkownik wybiera jedną grupę laboratoryjną, dla której są przeznaczone dane zajęcia. Pole Grupa 2 jest wyłączone.

## Formularz dodawania zajęć typu Projekt



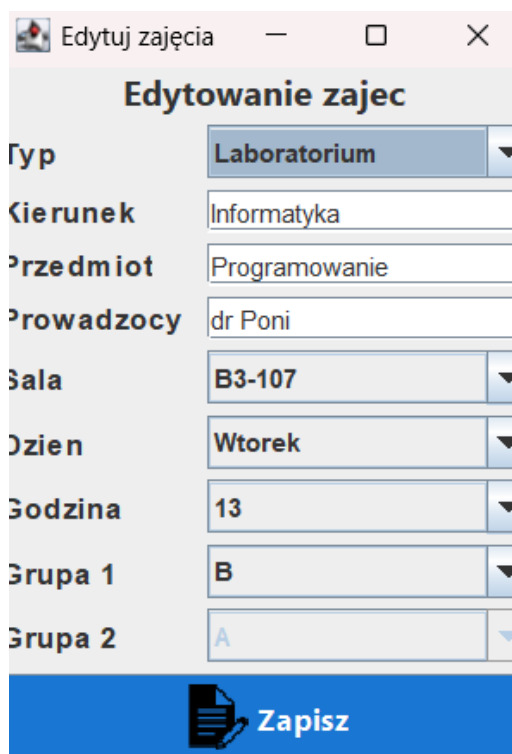
Typ	Projekt
Kierunek	
Przedmiot	
Prowadzocy	
Sala	B1-201
Dzień	Poniedziałek
Godzina	9
Grupa 1	A
Grupa 2	B

**+ Zapisz**

Rys. 4.5. Formularz dodawania zajęć typu Projekt

Dla zajęć typu Projekt dostępne są dwa pola wyboru grup: Grupa 1 i Grupa 2. System wymusza wybranie obu, ponieważ projekty realizowane są wspólnie przez dwie grupy.

## Formularz edycji zajęć



Edytuj zajęcia

**Edytowanie zajęć**

Typ: Laboratorium

Kierunek: Informatyka

Przedmiot: Programowanie

Prowadzący: dr Poni


Sala: B3-107

Dzień: Wtorek

Godzina: 13

Grupa 1: B

Grupa 2: A

 **Zapisz**

Rys. 4.6. Formularz edycji istniejących zajęć

Formularz edycji pozwala na modyfikację danych wcześniej zapisanych zajęć. Po zaznaczeniu wiersza w tabeli, dane są automatycznie ładowane do formularza. Po kliknięciu **Zapisz**, rekord zostaje zaktualizowany w bazie danych. System waliduje dane przed zapisem.

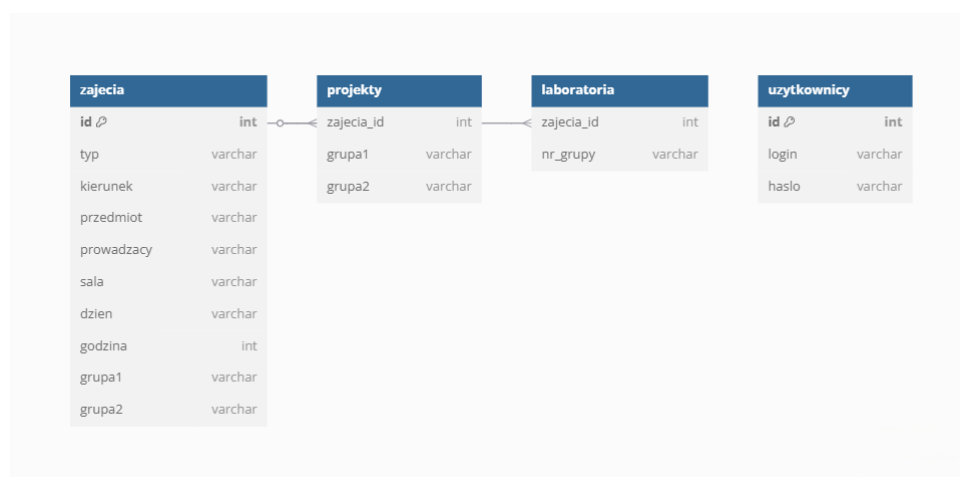
## 4.1. Baza danych

Dane przechowywane są w relacyjnej bazie danych **PostgreSQL**, w ramach schematu `public` bazy `javabase`. System korzysta z następujących tabel:

- `zajecia` – główna tabela przechowująca dane o wszystkich zajęciach:
  - `id`, `typ`, `kierunek`, `przedmiot`, `prowadzacy`, `sala`, `dzien`, `godzina`, `grupa1`, `grupa2`.
- `projekty` – zawiera przypisanie dwóch grup do zajęć typu Projekt:
  - `zajecia_id`, `grupa1`, `grupa2`.
- `laboratoria` – przechowuje numer grupy przypisanej do zajęć typu Laboratorium:
  - `zajecia_id`, `nr_grupy`.
- `uzytkownicy` – tabela logowania przechowująca dane uwierzytelniające użytkowników:
  - `id`, `login`, `haslo`.

Tabele są powiązane logicznie przez kolumnę `zajecia_id`, a dane zabezpieczone są poprzez ograniczenia integralności oraz indeksy. Struktura została zaprojektowana tak, aby umożliwiać wygodne wykonywanie operacji CRUD i filtrowania.

## 4.2. Diagram Baza Danych



**Rys. 4.7.** Diagram relacyjny bazy danych systemu

## 4.3. Interfejs użytkownika

Interfejs został wykonany w technologii **Swing**. Główne okna aplikacji to:

- **Ekran logowania** – umożliwia dostęp tylko zalogowanym użytkownikom;
- **Panel sekretariatu** – pozwala na przeglądanie i zarządzanie zajęciami;
- **Formularz dodawania zajęć** – umożliwia wprowadzenie nowych zajęć;
- **Formularz edycji zajęć** – pozwala na modyfikację istniejących rekordów.

## 4.4. Walidacja i błędy

System zawiera zabezpieczenia:

- Sprawdzanie dostępności sali w wybranym dniu i godzinie;
- Sprawdzanie, czy dana grupa nie ma już zajęć w tym czasie;
- Obsługa wyjątków SQL i wyświetlanie komunikatów błędów użytkownikowi.

## 4.5. Szczegółowy opis komponentów GUI

### 4.5.1. Formularz Dodawania Zajęć (DodajZajeciaPanel)

- **Cel:** Dodawanie nowych rekordów zajęć do bazy danych.
- **Elementy:**
  - `JComboBox` – typ zajęć (np. Wykład, Laboratorium, Projekt);
  - `TextField` – kierunek, przedmiot, prowadzący, sala, godzina, grupa1, grupa2;
  - `JComboBox` – dzień tygodnia;
  - `Button` – „Zapisz” — zapisuje dane do bazy danych.
- **Uwagi:** Interfejs zawiera etykiety (`JLabel`) przypisane do każdego pola. Formularz obsługuje walidację przed zapisem zajęć.

### 4.5.2. Formularz Edytowania Zajęć (EdytujZajeciaPanel)

- **Cel:** Edytowanie istniejących danych zajęć.
- **Elementy:** Te same co w `DodajZajeciaPanel`, jednak służą do aktualizacji danych:
  - Pola są wstępnie wypełnione danymi z wybranego rekordu;
  - `Button` – „Zapisz” — aktualizuje dane w bazie.
- **Uwagi:** Pola są edytowalne i automatycznie uzupełniane na podstawie danych wybranych z tabeli.

### 4.5.3. Formularz Logowania (Login)

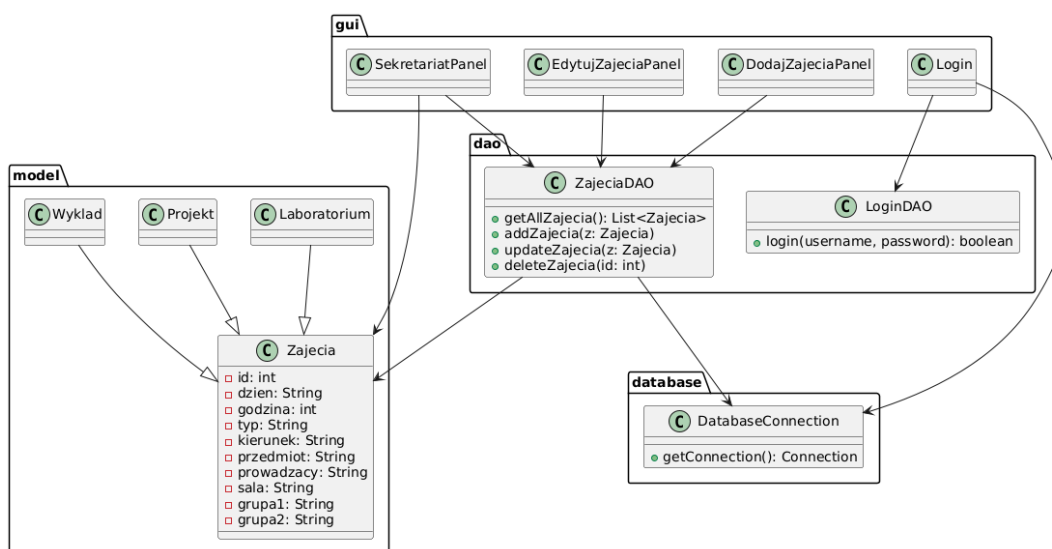
- **Cel:** Autoryzacja użytkownika w systemie.
- **Elementy:**
  - JTextField – login;
  - JPasswordField – hasło;
  - JButton – „Zaloguj” — weryfikuje dane logowania.
- **Uwagi:** Prosty, nowoczesny wygląd z ikoną kłódki; Zabezpieczenie dostępu do aplikacji.

### 4.5.4. Panel Sekretariatu (SekretariatPanel)

- **Cel:** Zarządzanie zajęciami (CRUD).
- **Elementy:**
  - JTable – wyświetlanie listy zajęć;
  - JComboBox – filtrowanie po sali, grupie, typie zajęć;
  - JButton – „Dodaj”, „Usuń”, „Edytuj”, „Zastosuj filtry”, „Odśwież”;
  - JScrollPane – przewijana tabela;
  - JLabel – etykiety opisowe.
- **Uwagi:** Obsługuje filtrowanie i pełną obsługę CRUD; główny ekran zarządzania.

## 4.6. Diagram komponentów GUI i zależności

Diagram komponentów GUI i zależności



Rys. 4.8. Zależności pomiędzy komponentami GUI, modelem danych i warstwą DAO

# Spis rysunków

4.1	Panel logowania do systemu . . . . .	5
4.2	Główny panel sekretariatu z listą zajęć i filtrowaniem . . . . .	6
4.3	Formularz dodawania zajęć typu Wykład . . . . .	7
4.4	Formularz dodawania zajęć typu Laboratorium . . . . .	8
4.5	Formularz dodawania zajęć typu Projekt . . . . .	9
4.6	Formularz edycji istniejących zajęć . . . . .	10
4.7	Diagram relacyjny bazy danych systemu . . . . .	11
4.8	Zależności pomiędzy komponentami GUI, modelem danych i warstwą DAO . . . . .	13



## **Spis tabel**

## **Spis listingów**

Załącznik nr 2 do Zarządzenia nr 228/2021 Rektora Uniwersytetu Rzeszowskiego z dnia 1 grudnia 2021 roku w sprawie ustalenia procedury antyplagiatowej w Uniwersytecie Rzeszowskim

## OŚWIADCZENIE STUDENTA O SAMODZIELNOŚCI PRACY

.....Mykhailo Kleban.....  
Imię (imiona) i nazwisko studenta

Wydział Nauk Ścisłych i Technicznych

.....Informatyka.....  
Nazwa kierunku

.....134922.....  
Numer albumu

1. Oświadczam, że moja praca projektowa pt.: System rezerwacji sal/podział godzin
  - 1) została przygotowana przeze mnie samodzielnie\*,
  - 2) nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2021 r., poz. 1062) oraz dóbr osobistych chronionych prawem cywilnym,
  - 3) nie zawiera danych i informacji, które uzyskałem/am w sposób niedozwolony,
  - 4) nie była podstawą otrzymania oceny z innego przedmiotu na uczelni wyższej ani mnie, ani innej osobie.
2. Jednocześnie wyrażam zgodę/nie wyrażam zgody\*\* na udostępnienie mojej pracy projektowej do celów naukowo-badawczych z poszanowaniem przepisów ustawy o prawie autorskim i prawach pokrewnych.

\_\_\_\_\_  
(miejscowość, data)

\_\_\_\_\_  
(czytelny podpis studenta)

\* Uwzględniając merytoryczny wkład prowadzącego przedmiot

\*\* – niepotrzebne skreślić