

Sprawozdanie z projektu  
realizowanego w ramach  
przedmiotu  
„Metody tworzenia  
aplikacji bazodanowych”

**Piotr Rzewnicki**

Kierownik przedmiotu:  
dr inż. Maciej Stodolski

Warszawa 2020

Oświadczam, że niniejsza praca stanowiąca podstawę do uznania osiągnięcia efektów uczenia się została wykonana przeze mnie samodzielnie.

## Spis treści

<b>1. Wstęp .....</b>	<b>5</b>
<b>1.1 Wymagania do uruchomienia aplikacji .....</b>	<b>5</b>
<b>1.2 Wykorzystanie zewnętrznych usług .....</b>	<b>5</b>
<b>2. Budowa aplikacji .....</b>	<b>6</b>
<b>2.1 Architektura programu .....</b>	<b>6</b>
<b>2.2 Diagram klas aplikacji .....</b>	<b>7</b>
<b>2.3 Budowa bazy danych .....</b>	<b>8</b>
<b>2.4 Komunikacja z bazą danych .....</b>	<b>8</b>
<b>2.5 Tworzenie dokumentów PDF .....</b>	<b>9</b>
<b>3. Architektura techniczna .....</b>	<b>10</b>
<b>4. Opis aplikacji .....</b>	<b>10</b>
<b>4.1 Widok książek.....</b>	<b>10</b>
<b>4.2 Widok czytelników .....</b>	<b>12</b>
<b>4.3 Widok wypożyczeń .....</b>	<b>13</b>
<b>4.4 Widok raportów .....</b>	<b>15</b>
<b>4.5 Filtrowanie tabel.....</b>	<b>16</b>
<b>5. Opis spełnienia wymagań z zakresu .....</b>	<b>17</b>
<b>5.1 Wymaganie W001 .....</b>	<b>17</b>
<b>5.2 Wymaganie W002 .....</b>	<b>17</b>
<b>5.3 Wymaganie W003 .....</b>	<b>17</b>
<b>5.4 Wymaganie W004 .....</b>	<b>18</b>
<b>5.5 Wymaganie W005 .....</b>	<b>18</b>
<b>5.6 Wymaganie W006 .....</b>	<b>18</b>
<b>5.7 Wymaganie W007 .....</b>	<b>19</b>
<b>5.8 Wymaganie W008 (brak realizacji) .....</b>	<b>19</b>
<b>5.9 Wymaganie W009 .....</b>	<b>19</b>
<b>5.10 Wymaganie W010 .....</b>	<b>20</b>
<b>5.11 Wymaganie W011 .....</b>	<b>20</b>
<b>6. Dalsze możliwości rozwoju .....</b>	<b>21</b>
<b>6.1 Wykupienie serwera z większą pojemnością bazy danych.....</b>	<b>21</b>
<b>6.2 Dodanie panelu logowania.....</b>	<b>21</b>
<b>6.3 Dodanie publicznie dostępnego panelu dla czytelników .....</b>	<b>21</b>

<b>6.4 Dodanie informacji o lokalizacji książki .....</b>	<b>21</b>
<b>6.6 Poprawa graficznego interfejsu użytkownika .....</b>	<b>21</b>
<b>6.7 Ochrona przed atakami SQL Injection.....</b>	<b>22</b>
<b>6.5 Generowanie innych dokumentów .....</b>	<b>22</b>
<b>6.8 Implementacja komunikacji z czytnikiem kodów .....</b>	<b>22</b>
<b>Dodatek A. Przykładowy ranking czytelnictwa .....</b>	<b>23</b>
<b>Dodatek B. Przykładowa lista zbyt długo wypożyczonych książek .....</b>	<b>24</b>
<b>Bibliografia .....</b>	<b>25</b>

# 1. Wstęp

W dzisiejszych czasach technologia rozwija się w sposób bardzo dynamiczny. W codziennym życiu spotykamy się z komputerami na każdym kroku. Jednak istnieją jeszcze miejsca, gdzie mimo powszechnej informatyzacji komputery nie są wykorzystywane do ułatwiania pracy.

Zauważyłem, że takimi miejscami są biblioteki, zwłaszcza w mniejszych szkołach, miejscowościach, gdzie dane czytelników, wypożyczenia książek są zapisywane na kartach bibliotecznych. Jest to bardzo mało efektywne, ponieważ pracownik biblioteki, aby zanotować wypożyczenie książek musi znaleźć konkretną kartę czytelnika pośród wielu innych kart. Kolejną wadą takiego rozwiązania jest możliwość uszkodzenia kart czytelników (zalanie, spalenie, wyblaknięcie tuszu). W takim wypadku biblioteka nie ma możliwości sprawdzenia, który kto wypożyczył daną książkę. Może to prowadzić do przywłaszczenia książki przez czytelnika.

Powyższe przemyślenia wpłynęły na fakt, że w ramach projektu zdecydowałem się stworzyć aplikację do zarządzania biblioteką. Aplikacja umożliwia wykonywanie podstawowych działań – rejestrowanie nowych czytelników, dodawanie książek do zbioru, notowanie wypożyczenia utworów. Program umożliwia również generowanie rankingu czytelników, co może zachęcić czytelników do większej aktywności. Tworzona jest również list osób, które mają wypożyczoną książkę dłużej niż dopuszczalny okres, który wynosi 14 dni.

## 1.1 Wymagania do uruchomienia aplikacji

Aby uruchomić aplikację, wymagane jest zainstalowane na komputerze środowisko uruchomieniowe dla programów napisanych w języku Java – Java Runtime Environment (JRE).

## 1.2 Wykorzystanie zewnętrznych usług

Aplikacja korzysta ze zdalnej bazy danych udostępnionej za pomocą serwisu [www.freesqldatabase.com](http://www.freesqldatabase.com). Istnieje możliwość zastąpienia zdalnej bazy danych lokalną.

## 2. Budowa aplikacji

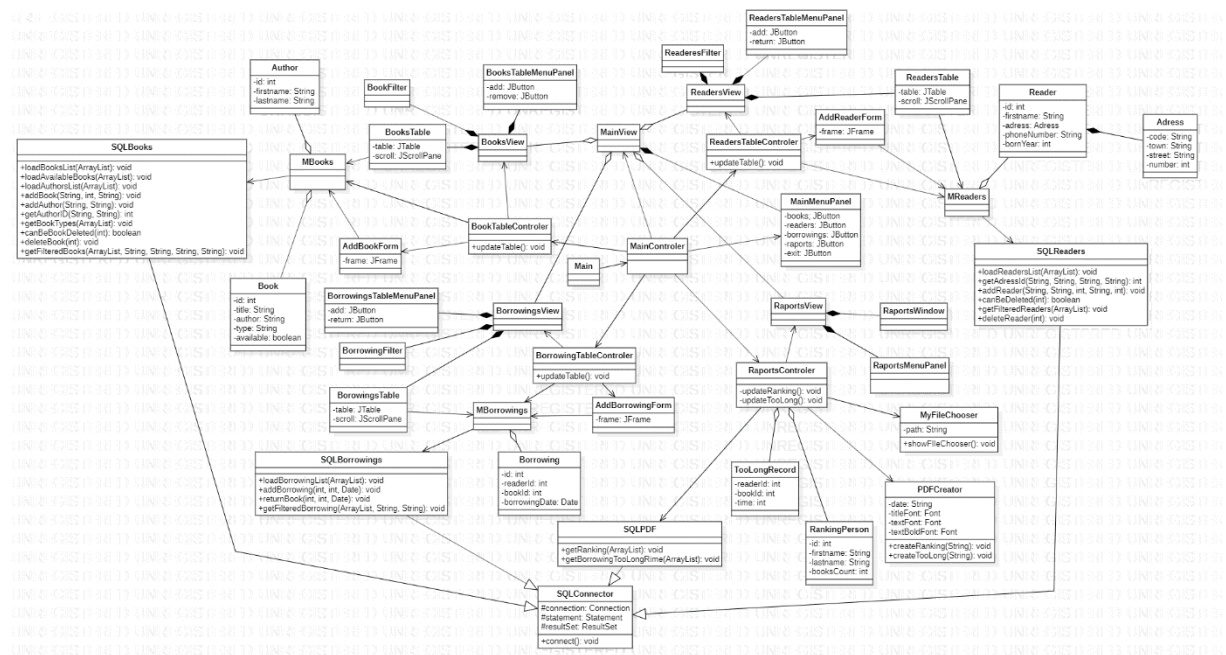
### 2.1 Architektura programu

Aplikacja składa się z kilkudziesięciu klas, które podzielone są na pakietu. Omówię krótko jakie zadanie mają klasy znajdujące się w poszczególnych pakietach:

- „app” – zawiera klasę służącą do uruchomienia aplikacji;
- „controlers” – zawiera klasy będące kontrolerami poszczególnych widoków;
- „filters” – zawiera klasy będące widokami okien filtrowania tabeli;
- „formsvIEWS” – zawiera klasy będące formularzami służącymi do dodawania książek, czytelników i wypożyczeń;
- „mdata” – zawiera klasy przechowujące dane (modele we wzorcu MVC);
- „pdfcreator” – zawiera klasę generującą dokumenty PDF;
- „sql” – zawiera klasy odpowiedzialne za komunikację z bazą danych;
- „view” – zawiera klasy z widokami.

## 2.2 Diagram klas aplikaciji

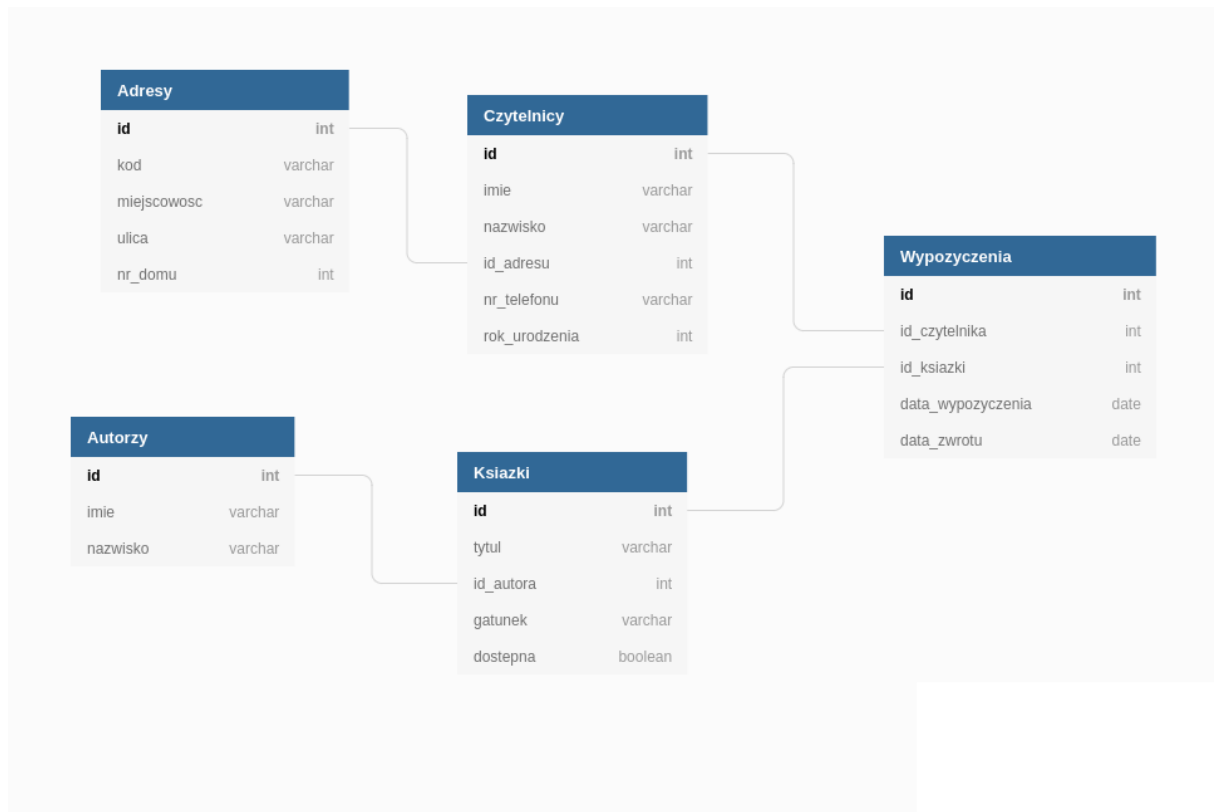
Poniższy diagram klas przedstawia wyłącznie główne atrybuty i metody poszczególnych klas, a więc pominięte zostały tzw. „gettery” i „settery”. W klasach odpowiedzialnych za widok, w celu zachowania większej czytelności, zostały pominięte mało istotne elementy graficzne np. etykiety, pola do wpisywania wartości.



*Rysunek 1: Diagram klas aplikacji*

## 2.3 Budowa bazy danych

Baza danych składa się z 5 tabel (rysunek nr 2), w których przechowywane są informacje na temat czytelników, książek i wypożyczeń.



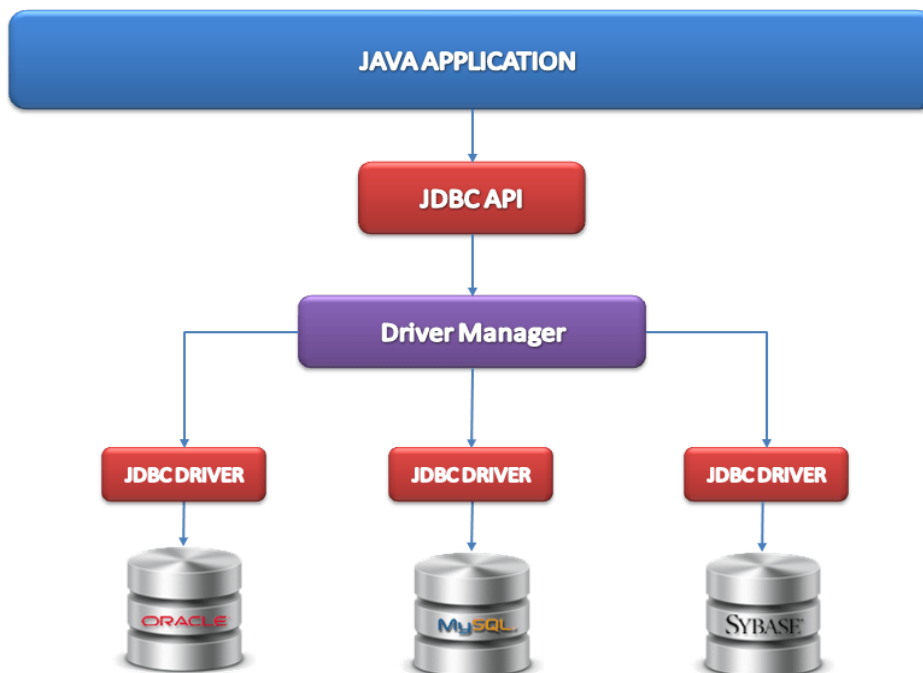
Rysunek 2: Architektura bazy danych

## 2.4 Komunikacja z bazą danych

Aplikacja komunikuje się z bazą danych za pomocą JDBC. Jest to interfejs umożliwiający połączenie się z bazą danych z poziomu aplikacji napisanej w Javie (rysunek nr 3). Składa się z 3 podstawowych elementów: interfejsu „Connection”, interfejsu „Statement” oraz interfejsu „ResultSet”<sup>[1]</sup>.

Interfejs „Connection” reprezentuje połączenie z bazą danych. Za pomocą interfejsu „Statement” można wykonywać operacje na bazie danych przy użyciu języka SQL. Interfejs „ResultSet” reprezentuje tabelę z pobranymi danymi z bazy, która jest wynikiem zapytania wysłanego za pomocą interfejsu „Statement”.





Rysunek 3: Komunikacja z bazą danych za pomocą JDBC  
źródło: [www.avaldes.com/connecting-to-sqlserver-using-jdbc/](http://www.avaldes.com/connecting-to-sqlserver-using-jdbc/)

## 2.5 Tworzenie dokumentów PDF

Dokumenty PDF generowane są za pomocą biblioteki „iText”. Jest to otwarta biblioteka, która umożliwia tworzenie i edytowanie dokumentów w formacie PDF, w języku Java.

### 3. Architektura techniczna

Do tworzenia aplikacji użyłem zintegrowanego środowiska programistycznego (IDE) NetBeans. Wybrałem to środowisko ze względu na bardzo dobrą jego znajomość, co znacznie przyspieszyło pracę nad aplikacją. Znajomość środowiska programistycznego bardzo ułatwia pracę np. przez automatyczne tworzenie metod „get”, „set” i konstruktora klasy.

### 4. Opis aplikacji

Aplikacja składa się z czterech głównych widoków: książek, czytelników, wypożyczeń i raportów.

Pomiędzy widokami można przełączać się za pomocą głównego menu znajdującego się w górnej części widoku aplikacji. Widoki posiadają swoje podmenu znajdujące się w prawej części okna głównego programu.

#### 4.1 Widok książek

Widok zarządzania książkami (rysunek nr 4) przedstawia tabelę z książkami znajdującymi się w bibliotece. Tabela zawiera utwory posortowane alfabetycznie.

Każda książka opisana jest parametrami:

- ID książki,
- tytuł,
- autor,
- gatunek książki,
- status książki.

W bocznym panelu menu Użytkownik ma możliwość zarządzania księgozbiorem znajdującym się w bibliotece.

Po wybraniu przycisku „Dodaj” wyświetli się formularz (rysunek nr 5), który należy wypełnić w celu dodania nowej książki. Autor jest wybierany z listy rozwijanej, na której znajdują się autorzy, których dzieła dostępne są w bibliotece. W przypadku, gdy autora nie ma na liście, po wybraniu opcji „Inny” pojawi się formularz, w który należy wpisać imię i nazwisko autora.

Po wybraniu przycisku „Usuń” zostanie usunięta zaznaczona książka na liście. Usunięcie książki możliwe jest wyłącznie, gdy znajduje się ona w bibliotece (status książki ma wartość „Dostępna”).

Biblioteka				
Książki		Czytelnicy	Wypożyczenia	Raporty
Tytuł: <input type="text"/>		Autor: <input type="text"/>	Gatunek: <input type="text"/>	Status: <input type="text"/>
		Szukaj		
ID	Tytuł	Autor	Gatunek	Status książki
36	Aimeiz	Piotr Bronikowski	Fantastyka	Wypożyczona
37	Alicja	Jacek Piekara	Fantastyka	Wypożyczona
38	Bot	Maksym Kidruk	Fantastyka	Dostępna
40	Czarne koby	Edgar Hyniewicz	Fantastyka	Dostępna
41	Czas próby	David Weber	Fantastyka	Wypożyczona
23	Folwark Zwierzęcy	George Orwell	Powieść	Wypożyczona
33	Głód	Graham Masterton	Horror	Wypożyczona
32	Grób	Max Czornyj	Horror	Dostępna
15	Harry Potter i Insignia Śmierci	Joanne Rowling	Literatura młodzieżowa	Dostępna
2	Harry Potter i Kamień Filizoficzny	Joanne Rowling	Literatura młodzieżowa	Wypożyczona
5	Harry Potter i Komnata Tajemnic	Joanne Rowling	Literatura młodzieżowa	Wypożyczona
13	Harry Potter i Książę Półkrwi	Joanne Rowling	Literatura młodzieżowa	Wypożyczona
6	Harry Potter i Wieża Azkabanu	Joanne Rowling	Literatura młodzieżowa	Wypożyczona
8	Język C++	Stephen Prata	Programowanie	Dostępna
18	krótka historia czasu	Stephen Hawking	Popularyzacja naukowa	Wypożyczona
31	Kryjówka	Nora Roberts	Kryminał	Dostępna
1	Krzyżacy	Henryk Sienkiewicz	Historyczna	Wypożyczona
21	Księża Jakubowe	Olga Tokarczuk	Powieść	Wypożyczona
20	Lśnienie	Stephen King	Powieść	Wypożyczona
24	Magazyn	Rob Hart	Powieść	Wypożyczona
34	Miasteczko Salem	Stephen King	Horror	Wypożyczona
25	My	Jewgienij Zamiatin	Fantastyka	Wypożyczona
35	Ocalić przyszłość	Ireneusz Osłowski	Fantastyka	Wypożyczona
10	Quo vadis	Henryk Sienkiewicz	Historyczna	Wypożyczona
17	Rok 1984	George Orwell	Antyutopia	Wypożyczona
19	Słońce też wschodzi	Ernest Hemingway	Powieść	Wypożyczona
30	Sodoma	Marcin Wolski	Fantastyka	Wypożyczona
16	Stary człowiek i morze	Ernest Hemingway	Opowiadanie	Wypożyczona
22	Szafa	Olga Tokarczuk	Opowiadanie	Wypożyczona
7	Thinking in Java	Bruce Eckel	Programowanie	Dostępna
3	W pustyni i w puszczy	Henryk Sienkiewicz	Powieść przygodowa	Wypożyczona
9	Wesele	Stanisław Wyspiański	Dramat	Wypożyczona
26	Wieczna wojna	Joe Haldeman	Fantastyka	Wypożyczona
39	Wiedźmin	Andrzej Sapkowski	Fantastyka	Wypożyczona
27	Wrona	Margaret Owen	Fantastyka	Wypożyczona
29	Wybrana	Naomi Novik	Fantastyka	Wypożyczona

Rysunek 4: Widok okna zarządzania książkami

Dodaj książkę

Tytuł:

Gatunek:

Autor:

Andrzej Sapkowski

Anuluj

Dodaj

Dodaj książkę

Tytuł:

Gatunek:

Autor:

Inny

Imię autora:

Nazwisko autora:

Anuluj

Dodaj

Rysunek 5: Widok formularza dodania książki: z wybranym autorem (po lewej) i z wybraną opcją "Inny" (po prawej)

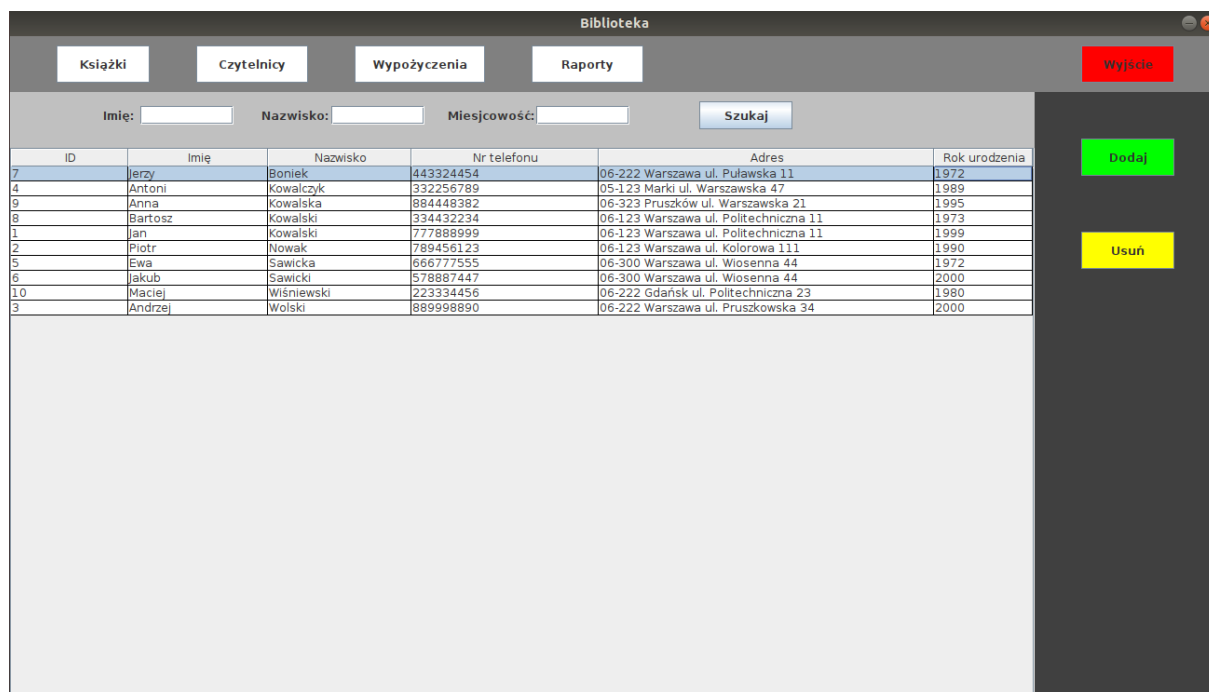
## 4.2 Widok czytelników

Okno zarządzania czytelnikami (rysunek nr 6) składa się z tabeli zawierającej listę czytelników zarejestrowanych w bibliotece. Czytelnicy ułożeni są na liście według kolejności alfabetycznej.

Każdy czytelnik opisany jest parametrami:

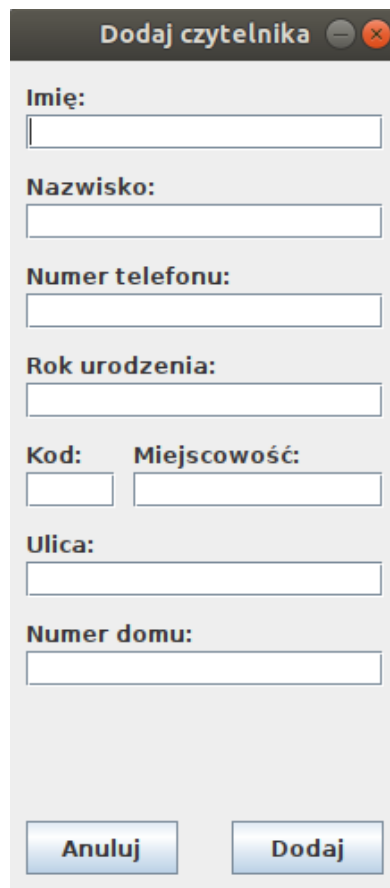
- ID czytelnika,
- imię,
- nazwisko,
- numer telefonu,
- adres (kod pocztowy, miejscowość, ulica, numer domu),
- rok urodzenia.

Za pomocą bocznego menu można zarządzać czytelnikami. Aby dodać czytelnika, należy wypełnić formularz zawierający jego podstawowe dane (rysunek nr 7). Usuwanie czytelnika jest możliwe wyłącznie, gdy nie ma on żadnych wypożyczonych książek.



ID	Imię	Nazwisko	Nr telefonu	Adres	Rok urodzenia
7	Jerzy	Boniek	443324454	06-222 Warszawa ul. Puławska 11	1972
4	Antoni	Kowalczyk	332256789	05-123 Marki ul. Warszawska 47	1989
9	Anna	Kowalska	884448382	06-323 Pruszków ul. Warszawska 21	1995
8	Bartosz	Kowalski	334432234	06-123 Warszawa ul. Politechniczna 11	1973
1	Jan	Kowalski	777888999	06-123 Warszawa ul. Politechniczna 11	1999
2	Piotr	Nowak	789456123	06-123 Warszawa ul. Kolorowa 111	1990
5	Ewa	Sawicka	666777555	06-300 Warszawa ul. Wiosenna 44	1972
6	Jakub	Sawicki	578887447	06-300 Warszawa ul. Wiosenna 44	2000
10	Maciej	Wiśniewski	223334456	06-222 Gdańsk ul. Politechniczna 23	1980
3	Andrzej	Wolski	889998890	06-222 Warszawa ul. Pruszkowska 34	2000

Rysunek 6: Widok zarządzania czytelnikami



Rysunek 7: Okno dodania nowego czytelnika

### 4.3 Widok wypożyczeń

W tym oknie (rysunek nr 8) znajduje się lista przedstawiająca aktualnie wypożyczone książki w kolejności od najstarszego wypożyczenia do najnowszego.

Lista opisana jest parametrami:

- ID wypożyczającego,
- wypożyczający (imię i nazwisko),
- ID książki,
- tytuł książki,
- data wypożyczenia.

Boczne menu umożliwia dodawanie nowych wypożyczeń (rysunek nr 9) oraz rejestrowanie zwrotów książek.

W oknie dodawania wypożyczeń z listy rozwijanej, zawierającej dostępne książki w bibliotece, wybieramy wypożyczaną książkę oraz z drugiej listy rozwijanej czytelnika, który wypożycza dany utwór. Istnieje również możliwość wyboru daty, kiedy książka została wypożyczona (domyślnie wpisywana jest aktualna data).

Po kliknięciu przycisku „Zwróć”, wypożyczenie zniknie z listy, a książka zmieni swój status z „Wypożyczona” na „Dostępna”. Jako data zwrotu książki zostanie zarejestrowana data kliknięcia w ten przycisk.

Książki

Czytelnicy

Wypożyczenia

Raporty

Wypożyczający:  Tytuł: 

Szukaj

ID wypożyczającego	Wypożyczający	ID książki	Tytuł książki	Data wypożyczenia
4	Antoni Kowalczyk	37	Alicja	2020-08-28
6	Jakub Sawicki	8	Harry Potter i Więzień Azkabanu	2020-09-16
1	Jan Kowalski	36	Aimeiz	2020-10-04
7	Jerzy Boniek	18	Krótką historią czasu	2020-10-08
5	Ewa Sawicka	29	Wybrana	2020-10-10
5	Ewa Sawicka	24	Magazyn	2020-10-19
9	Anna Kowalska	41	Czas próby	2020-10-20
1	Jan Kowalski	2	Harry Potter i Kamień Filizoficzny	2020-10-20
2	Piotr Nowak	21	Księgi Jakubowe	2020-10-21
3	Andrzej Wolski	20	Łńnienie	2020-10-22
3	Andrzej Wolski	5	Harry Potter i Komnata Tajemnic	2020-10-28
8	Bartosz Kowalski	23	Folwark Zwierzęcy	2020-10-28
2	Piotr Nowak	26	Wielka wojna	2020-10-29
4	Antoni Kowalczyk	9	Welese	2020-11-02
1	Jan Kowalski	30	Sodoma	2020-11-03
7	Jerzy Boniek	16	Stary człowiek i morze	2020-11-03
2	Piotr Nowak	35	Ocalić przyszłość	2020-11-03
4	Antoni Kowalczyk	17	Rok 1984	2020-11-03
2	Anna Kowalska	25	My	2020-11-04
9	Piotr Nowak	1	Krzyżacy	2020-11-06
6	Jakub Sawicki	3	W pustyni i w puszczy	2020-11-06
8	Bartosz Kowalski	10	Quo vadis	2020-11-07
7	Jerzy Boniek	28	Wysoki zamek	2020-11-09
1	Jan Kowalski	34	Miasteczko Salem	2020-11-10
9	Anna Kowalska	19	Słońce też wschodzi	2020-11-10
3	Andrzej Wolski	39	Wiedźmin	2020-11-10
4	Antoni Kowalczyk	33	Glód	2020-11-10
10	Maciej Wiśniewski	27	Wrona	2020-11-10
10	Maciej Wiśniewski	22	Szafa	2020-11-10

Wyjdź

Dodaj

Zwróć

Rysunek 8: Okno przedstawiające wypożyczone książki

Dodaj wypożyczenie

Książka:  
38 - Bot

Wypożyczający:  
7 - Boniek Jerzy

Data(RRRR-MM-DD):  
2020-11-11

AnulujDodaj

Rysunek 9: Okno wypożyczania książek

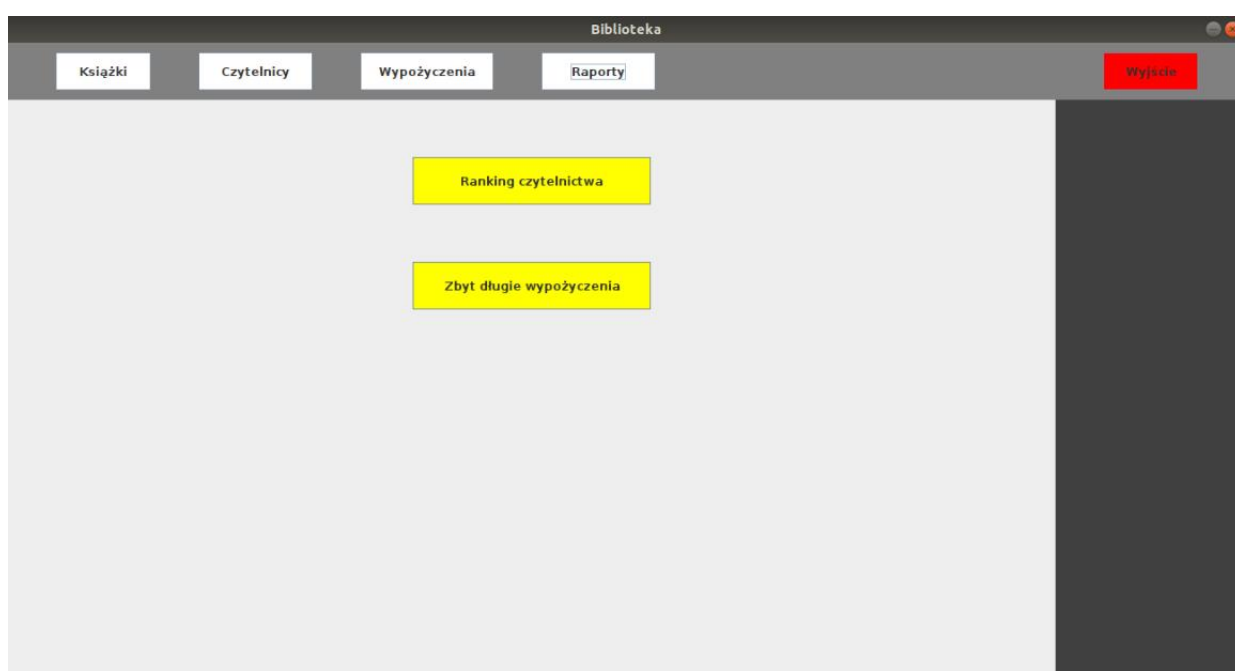
## 4.4 Widok raportów

W oknie generowania raportów (rysunek nr 10) znajdują się dwa przyciski:

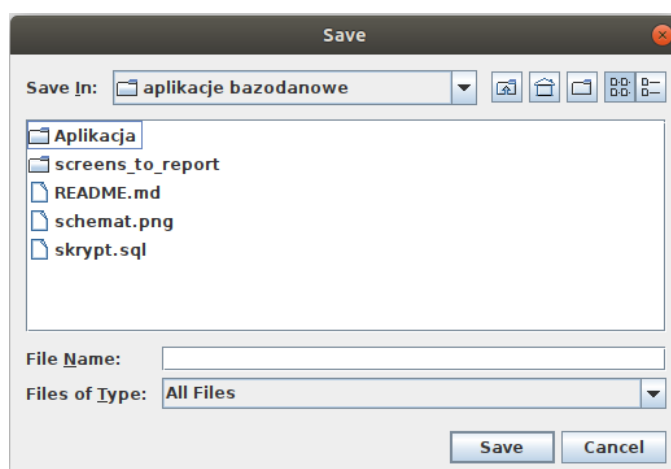
- „Ranking czytelnictwa”,
- „Zbyt długie wypożyczenia”.

Po wybraniu jednego z przycisków otworzy się okno wyboru lokalizacji pliku (rysunek nr 11), w którym należy wybrać miejsce zapisu raportu oraz wpisać jego nazwę.

Po zapisaniu utworzy się, w zależności od wybranego przycisku, ranking czytelnictwa z datą generowania raportu (przykładowy dokument jest w „Dodatku A”) lub listę osób zbyt długo wypożyczających książki („Dodatek B”) na dzień generowania pliku.



Rysunek 10: Widok okna generowania raportów



Rysunek 11: Okno wyboru lokalizacji pliku

## 4.5 Filtrowanie tabel

Użytkownik aplikacji ma możliwość filtrowania tabel z książkami, czytelnikami, wypożyczonymi zasobami, w celu szybszego znalezienia interesującego go elementu tabeli.

Filtrowanie tabel nie wymaga wpisywania pełnych nazw pól. Wpisanie tylko części nazwy umożliwia wyszukiwanie elementów, które zawierają ten fragment tekstu.

Przykładowo wpisanie w pole wyszukiwania przy nazwisku wartości „kował” znajdzie osoby mające w nazwisku ten fragment ciągu liter (np. „Kowalski”, „Kowalczyk”, „Kował” itp.).

Jak można zauważyć na przykładzie, wielkość liter również nie ma znaczenia na wyszukiwanie, tzn. wpisanie wartości „nowak” wyświetli takie same rezultaty jak wpisanie wartości „NOWAK”.

Przykładowy wynik filtrowania przedstawia rysunek nr 12.

The screenshot shows a web application titled "Biblioteka". At the top, there are four tabs: "Książki", "Czytelnicy", "Wypożyczenia", and "Raporty". The "Książki" tab is selected. Below the tabs, there is a search bar with the following fields: "Tytuł: potter", "Autor: Rowling", "Gatunek:" (with a dropdown arrow), "Status: Dostępna" (with a dropdown arrow), and a "Szukaj" button. To the right of the search bar, there is a red "Wyjście" button. Below the search bar, there is a table with the following data:

ID	Tytuł	Autor	Gatunek	Status książki
15	Harry Potter i Insygnia Śmierci	Joanne Rowling	Literatura młodzieżowa	Dostępna
13	Harry Potter i Książę Półkrwi	Joanne Rowling	Literatura młodzieżowa	Dostępna

Below the table, there is a large empty space. On the right side of the interface, there are two buttons: a green "Dodaj" button and a yellow "Usuń" button.

Rysunek 12: Widok tabeli po użyciu filtru



## **5. Opis spełnienia wymagań z zakresu**

### **5.1 Wymaganie W001**

#### **Treść wymagania:**

„Aplikacja ma działać na dowolnym systemie operacyjnym (Windows, Linux).”

#### **Opis realizacji:**

Aplikacja została napisana w języku Java, a więc jest uruchamiana w JVM (Java Virtual Machin), co powoduje jednakowe zachowanie programu bez względu na system operacyjny w jakim jest uruchamiany.

### **5.2 Wymaganie W002**

#### **Treść wymagania:**

„Aplikacja ma umożliwiać użytkownikowi dodawanie nowych książek do bazy i wyświetlanie listy wszystkich pozycji.”

#### **Opis realizacji:**

Aplikacja wyświetla wszystkie książki znajdujące się w bibliotece i umożliwia dodawanie nowych. Szczegóły realizacji wymagania znajdują się w rozdziale „4.1 Widok książek”.

### **5.3 Wymaganie W003**

#### **Treść wymagania:**

„Każda książka powinna być opisana przynajmniej parametrami: tytuł, autor, nr identyfikacyjny, gatunek, status książki (wypożyczona, w bibliotece).”

#### **Opis realizacji:**

Każda książka znajdująca się w bibliotece opisana jest powyższymi parametrami. Status książki przyjmuje wartości: „Wypożyczona” i „Dostępna”. Szczegółowe przedstawienie realizacji wymagania znajduje się w rozdziale „4.1 Widok książek”.

## **5.4 Wymaganie W004**

### **Treść wymagania:**

„Aplikacja powinna umożliwiać wyszukiwanie książek wg. nazwy, gatunku, autora.”

### **Opis realizacji:**

Użytkownik ma możliwość filtrowania tabel z książkami według powyższych parametrów. Szczegółowy opis realizacji wymagania znajduje się w rozdziale „4.5 Filtrowanie tabel”.

## **5.5 Wymaganie W005**

### **Treść wymagania:**

„Użytkownik ma możliwość dodawania/edytowania/usuwania czytelników.”

### **Opis realizacji:**

Użytkownik za pomocą bocznego menu w oknie czytelników ma możliwość dodawania i usuwania czytelników. Szczegółowy opis znajduje się w rozdziale „4.2 Widok czytelników”.

Opcja edycji czytelników nie została zaimplementowana ze względu na bardzo rzadkie jej użycie. Równoznaczne z opcją edycji jest usunięcie i dodanie nowego czytelnika.

## **5.6 Wymaganie W006**

### **Treść wymagania:**

„Dane czytelnika powinny zawierać: imię, nazwisko, adres zamieszkania, nr telefonu, rok urodzenia.”

### **Opis realizacji:**

Każdy czytelnik zarejestrowany w bibliotece opisany jest za pomocą powyższych parametrów. Szczegółowe przedstawienie realizacji wymagania znajduje się w rozdziale „4.2 Widok czytelników”.

## **5.7 Wymaganie W007**

### **Treść wymagania:**

„Aplikacja ma rejestrować wypożyczenia książek (jaka książka, kto wypożyczył, kiedy została wypożyczona).”

### **Opis realizacji:**

Aplikacja umożliwia rejestrowanie wypożyczeń książek. Szczegółowe przedstawienie realizacji wymagania znajduje się w rozdziale „4.3 Widok wypożyczeń”.

## **5.8 Wymaganie W008 (brak realizacji)**

### **Treść wymagania:**

„Aplikacja musi generować raport o czytelniku.”

### **Przyczyna braku realizacji:**

Wymaganie nie zostało zrealizowane ze względu na fakt, że w tabeli zawierającej czytelników znajdują się wszystkie informacje o konkretnym czytelniku. Generowanie pliku PDF z danymi czytelnika nie ma praktycznego zastosowania, ponieważ znacznie szybsze jest odczytanie danych z tabelki.

## **5.9 Wymaganie W009**

### **Treść wymagania:**

„Aplikacja musi generować ranking czytelnictwa (posortowana lista czytelników wg. malejącej liczby wypożyczonych książek).”

### **Opis realizacji:**

Aplikacja umożliwia generowanie raportów. Instrukcja generowania raportów znajduje się w rozdziale „4.4 Widok raportów”. Przykładowy ranking czytelnictwa znajduje się w „Dodatku A”.

## **5.10 Wymaganie W010**

### **Treść wymagania:**

„Aplikacja musi generować raport zawierających czytelników, którzy mają zbyt długo wypożyczoną książkę (14 dni).”

### **Opis realizacji:**

Aplikacja umożliwia generowania raportów. Instrukcja generowania raportów znajduje się w rozdziale „4.4 Widok raportów”. Przykładowy raport znajduje się w „Dodatku B”.

## **5.11 Wymaganie W011**

### **Treść wymagania:**

„Aplikacja powinna działać dla dużej ilości książek/czytelników.”

### **Opis realizacji:**

Tabele z książkami i czytelnikami zawierają scroll, więc mogą być dowolnej długości.

Zdalna baza danych, z której korzysta aplikacja, ma maksymalną pojemność 5 MB, ale można ją łatwo zastąpić lokalną bazą danych, co umożliwi generowanie bardzo dużej ilości danych w aplikacji.

## **6. Dalsze możliwości rozwoju**

### **6.1 Wykupienie serwera z większą pojemnością bazy danych**

Aktualnie w darmowej wersji bazy danych dostępne jest 5 MB pamięci na dane. Jest to niewielka ilość, aby przechowywać duże ilości książek oraz kilkudziesięciu czytelników i listę wypożyczonych książek. Można skorzystać z bazy lokalnej na komputerze, ale przechowywanie danych zdalnie jest bezpieczniejsze i nie grozi utratą danych w przypadku awarii komputera.

### **6.2 Dodanie panelu logowania**

Kolejnym krokiem w rozwoju jest dodanie panelu do logowania po uruchomieniu aplikacji. Ma to na celu zabezpieczenie, że nikt nieuprawniony, przebywający w bibliotece, np. czytelnik, osoba sprzątająca, nie będzie miał dostępu do aplikacji co wiąże się również z dostępem do prywatnych danych czytelników.

### **6.3 Dodanie publicznie dostępnego panelu dla czytelników**

Dobrym kierunkiem w rozwoju aplikacji byłoby utworzenie publicznie dostępnego panelu, do którego mieliby dostęp czytelnicy po uruchomieniu aplikacji na komputerach znajdujących się w bibliotece. Mieliby możliwość wyszukiwania książek znajdujących się w bibliotece, co w łatwiejszy sposób umożliwi im sprawdzenie czy interesująca ich pozycja znajduje się w bibliotece.

### **6.4 Dodanie informacji o lokalizacji książki**

Informacje o książce należy rozszerzyć również o informację o jej położeniu w bibliotece, co po zaimplementowaniu poprzedniego rozwiązania usprawni proces wyszukiwania książek.

### **6.6 Poprawa graficznego interfejsu użytkownika**

Modyfikacji należy poddać graficzny interfejs użytkownika. W tabeli z listą wypożyczeń, należy wyróżnić (np. zaznaczając na czerwono) wypożyczenia, które trwają dłużej niż 14 dni.

## **6.7 Ochrona przed atakami SQL Injection**

Aplikację należy zabezpieczyć przed atakami SQL Injection. Aktualnie program nie posiada takiego zabezpieczenia ze względu, że adresatami aplikacji są bibliotekarze, którzy nie mają powodu, aby wykraść dane, ponieważ mają do nich dostęp. Ten krok należy wprowadzić, gdy zostanie utworzony publicznie dostępny panel dla czytelników.

## **6.5 Generowanie innych dokumentów**

Kolejną ciekawą funkcją, która pozytywnie wpłynie na odbiór aplikacji jest możliwość generowania raportów czytelnictwa dla każdego z czytelników. Raport taki zawierałby historię wypożyczeń książek przez konkretnego czytelnika.

Innym przydatnym dokumentem jest karta czytelnika zawierająca jego dane oraz numer identyfikacyjny zapisany w postaci kodu kreskowego.

Następnym dokumentem, który mógłby być generowany jest pismo informujące czytelnika, że czas wypożyczenia książki trwa dłużej niż 14 dni. Takie pismo byłoby dopasowane, aby po złożeniu pasowało do koperty z okienkiem, więc bibliotekarz nie będzie musiał pisać na każdej kopercie adresu.

## **6.8 Implementacja komunikacji z czytnikiem kodów**

Bardzo przydatnym narzędziem byłoby ustanowienie komunikacji aplikacji z czytnikiem kodów. Przyspieszyłoby to bardzo proces wypożyczania, który polegałby wyłącznie na zeskanowaniu kodu kreskowego z książki (zawierający jej numer ID) oraz z karty bibliotecznej czytelnika (z jego ID).

## **Dodatek A. Przykładowy ranking czytelnictwa**

### **Ranking czytelnictwa na dzień 10-11-2020**

<b>Numer ID</b>	<b>Imię</b>	<b>Nazwisko</b>	<b>Liczba książek</b>
2	Piotr	Nowak	7
8	Bartosz	Kowalski	6
1	Jan	Kowalski	6
4	Antoni	Kowalczyk	6
3	Andrzej	Wolski	5
5	Ewa	Sawicka	5
7	Jerzy	Boniek	3
9	Anna	Kowalska	3
6	Jakub	Sawicki	2
10	Maciej	Wiśniewski	2

## **Dodatek B. Przykładowa lista zbyt długo wypożyczonych książek**

### **Lista osób zbyt długo przetrzymujących książki na dzień 10-11-2020**

<b>Numer ID czytelnika</b>	<b>Imię</b>	<b>Nazwisko</b>	<b>Numer ID książki</b>	<b>Tytuł</b>	<b>Data wypożyczenia</b>	<b>Czas wypożyczenia [dni]</b>
4	Antoni	Kowalczyk	37	Alicja	2020-08-28	74
6	Jakub	Sawicki	6	Harry Potter i Wieżień Azkabanu	2020-09-16	55
1	Jan	Kowalski	36	Aimeiz	2020-10-04	37
7	Jerzy	Boniek	18	Krótką historia czasu	2020-10-08	33
5	Ewa	Sawicka	29	Wybrana	2020-10-10	31
5	Ewa	Sawicka	24	Magazyn	2020-10-19	22
1	Jan	Kowalski	2	Harry Potter i Kamień Filizoficzny	2020-10-20	21
9	Anna	Kowalska	41	Czas próby	2020-10-20	21
2	Piotr	Nowak	21	Księgi Jakubowe	2020-10-21	20
3	Andrzej	Wolski	20	Lśnienie	2020-10-22	19



## **Bibliografia**

[1] iProgramujesz.pl

<http://iprogramujesz.pl/kurs-jdbc-czesc-1/>