

# Projekt z programowania obiektowego

## Niezbędnik czytelnika

Daniela Tataruda

15 czerwca 2018

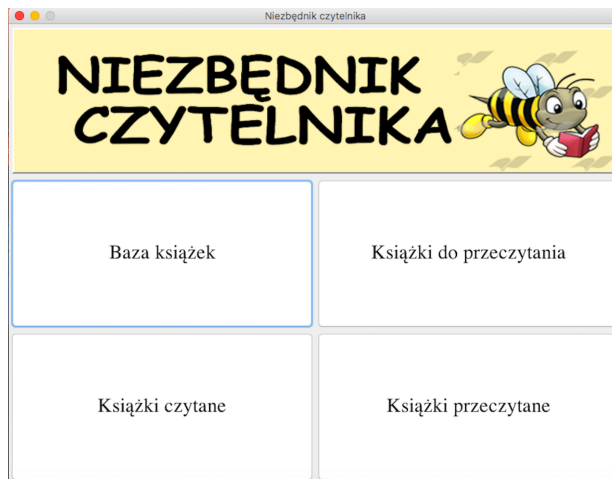
## 1 Wstęp

Niezbędnik czytelnika jest aplikacją okienkową napisaną w języku Java przy użyciu biblioteki Swing, która umożliwia obsługę interfejsu okienkowego. Aplikacja jest projektem semestralnym z programowania obiektowego na zajęcia w Instytucie Informatyki UWr. Program jest przeznaczony dla osób lubiących czytać i upamiętniać przeczytane przez siebie książki.

## 2 Opis programu

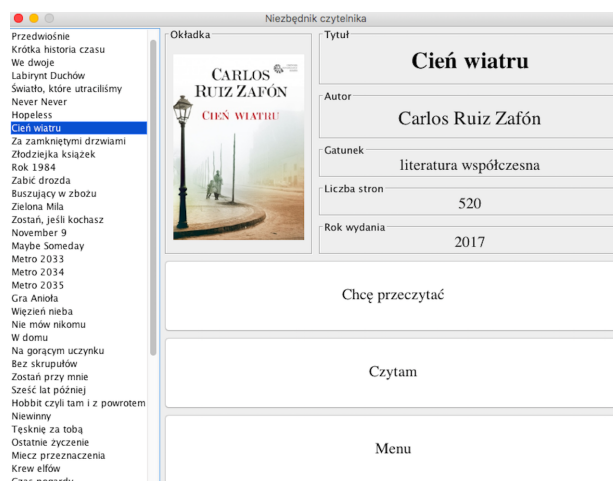
Aplikacja składa się z pięciu okien.

1. **Okno główne** - jest to podstawowe okno menu, zawiera logo aplikacji i cztery przyciski, które automatycznie po kliknięciu przenoszą do odpowiednich okien.



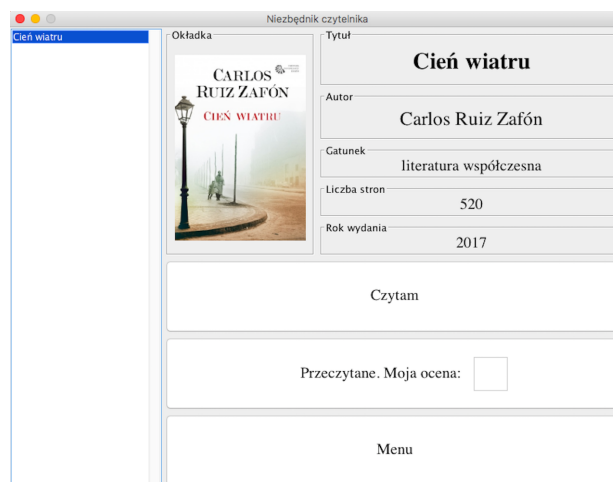
Rysunek 1: Okno Główne.

2. **Baza książek** - jest to okno, w którym wyświetlana jest cała podstawowa baza książek. Po prawej stronie wyświetlany jest interfejs zaznaczonej książki i panel z przyciskami pozwalający użytkownikowi wykonać trzy operacje (przeniesienie książki do bazy książek do przeczytania, zaznaczenie, że książka jest w chwili obecnej czytana oraz powrót do okna głównego).



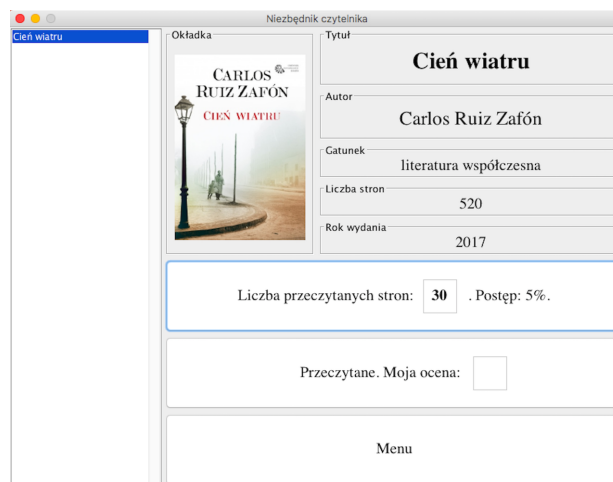
Rysunek 2: Baza książek.

3. **Baza książek do przeczytania** - jest to okno, w którym wyświetlana jest lista książek, które użytkownik zaznaczył jako te, które chce przeczytać. Po prawej stronie (tak jak w poprzednim oknie) wyświetlany jest interfejs zaznaczonej książki i panel z przyciskami pozwalający użytkownikowi wykonać trzy operacje (zaznaczenie, że książka jest w chwili obecnej czytana, ocenienie książki i przeniesienie do bazy książek przeczytanych oraz powrót do okna głównego).



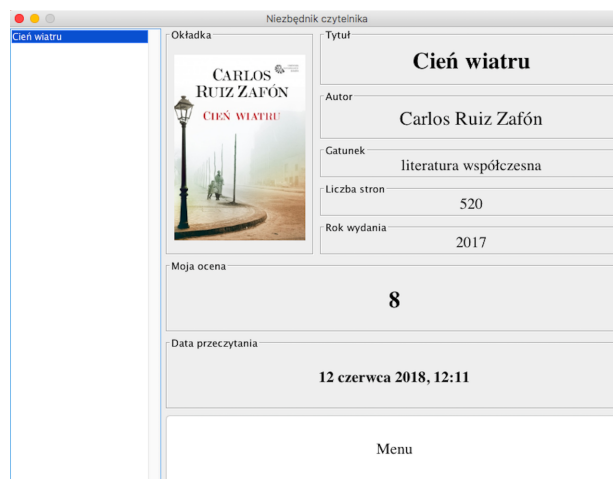
Rysunek 3: Baza książek do przeczytania.

4. **Baza książek czytanych** - jest to okno, w którym wyświetlana jest lista książek, które użytkownik zaznaczył jako te, które czyta. Po prawej stronie (tak jak w poprzednich oknach) wyświetlany jest interfejs zaznaczonej książki i panel z przyciskami pozwalający użytkownikowi wykonać trzy operacje (wprowadzenie liczby przeczytanych stron i zobaczenie jaki postęp w czytaniu został wykonany, ocenienie książki i przeniesienie do bazy książek przeczytanych oraz powrót do okna głównego).



Rysunek 4: Baza książek czytanych.

5. **Baza książek przeczytanych** - jest to okno, w którym wyświetlana jest lista książek, które użytkownik zaznaczył jako te, które przeczytał. Po prawej stronie (tak jak w poprzednich oknach) wyświetlany jest interfejs zaznaczonej książki i panele z informacjami o ocenie jaką wystawił użytkownik i data przeczytania danej książki. Na dole okna umieszczony jest przycisk, który służy do powrotu do okna głównego aplikacji.



Rysunek 5: Baza książek przeczytanych.

## 3 Analiza obiektowa

### 3.1 Lista klas z opisem

#### 3.1.1 Projekt

Klasa z jedną statyczną metodą `main()`, która uruchamia program.



**Rysunek 6:** Schemat klasy *Projekt*.

### 3.1.2 Okno

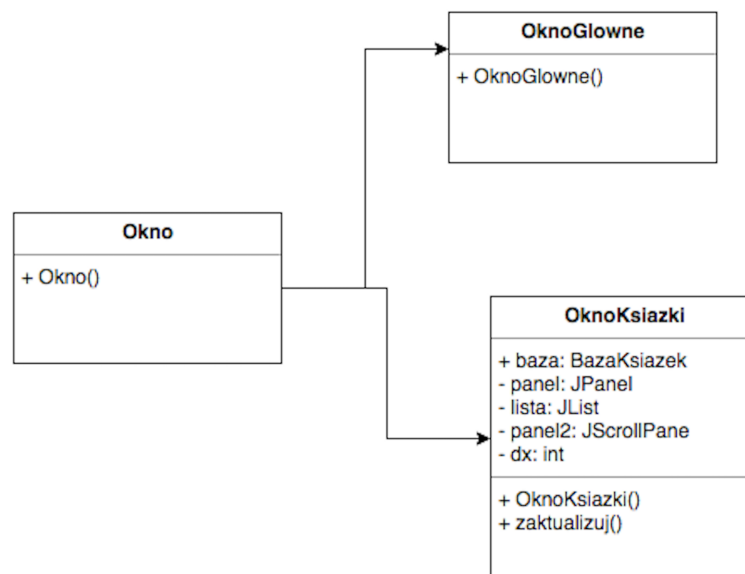
Klasa dziedzicząca po *JFrame*. Ustawia domyślne wartości dla okna (`resizable(false)`, wymiary, wyjście).

### 3.1.3 OknoGłowne

Klasa dziedzicząca po *Okno*. Kontroluje i ustawia wszystkie przyciski oraz obrazki pojawiające się w pierwszym podstawowym oknie aplikacji.

### 3.1.4 OknoKsiazki

Klasa dziedzicząca po *Okno*. Zawiera panel przesuwany z listą książek z bazy podanej jako parametr w konstruktorze oraz panel książki z klasy *PanelKsiazki*.



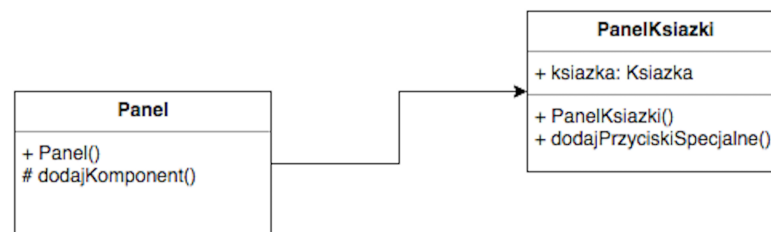
**Rysunek 7:** Schemat klas dziedziczących po *Okno*.

### 3.1.5 Panel

Klasa dziedzicząca po *JPanel*. Ma w sobie metodę *dodajKomponent* pozwalającą dodawać obiekty w podanych miejscach i z odpowiednimi parametrami podanymi jako argumenty funkcji.

### 3.1.6 PanelKsiazki

Klasa dziedzicząca po *Panel*. Jest to klasa ustawiająca cały interfejs książki znajdujący się w oknie książki, korzystająca z funkcji *dodajKomponent* i posiadająca metodę *dodajPrzyciskiSpecjalne* oraz ustawiająca przyciski do poruszania się po oknach.



Rysunek 8: Schemat klas dziedziczących po *Panel*.

### 3.1.7 Ksiazka

Klasa definiująca obiekt książki, czyli bazowy obiekt programu. Ma w sobie pola określające daną książkę oraz metody wydobywania tych pól, ponieważ są to pola chronione (nie dostępne z zewnątrz klasy). Zaimplementowana jest tutaj także metoda pobierająca okładkę z folderu *grafiki*.

### 3.1.8 KsiazkaDoPrzeczytania

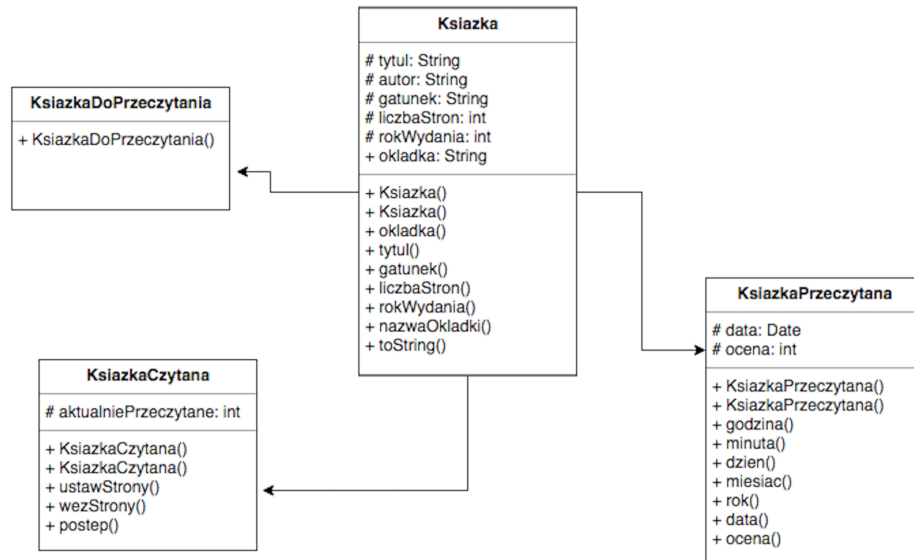
Klasa dziedzicząca po *Ksiazka* definiująca obiekt książki do przeczytania.

### 3.1.9 KsiazkaCzytana

Klasa dziedzicząca po *Ksiazka* definiująca obiekt książki czytanej. Posiada ona dodatkowe pole trzymające liczbę stron aktualnie przeczytanych i metodę obliczającą postęp w czytaniu.

### 3.1.10 KsiążkaPrzeczytana

Klasa dziedzicząca po *Książka* definiująca obiekt książki przeczytanej. Posiada ona dodatkowe pola oceny i daty oraz metodę pobierającą i wypisującą datę.



Rysunek 9: Schemat klas dziedziczących po *Książka*.

### 3.1.11 BazaKsiazek

Klasa trzymająca, czytająca i zapisująca bazę książek w postaci pliku z rozszerzeniem `.bin`. Implementuje interfejs *Serializable*. Baza jest przechowywana w postaci listy obiektów klasy *Książka*. Ma standardowe metody dodawania i usuwania elementów z listy oraz dwie metody obsługujące zapisywanie i czytanie z plików binarnych.

### 3.1.12 BazaDoPrzeczytania

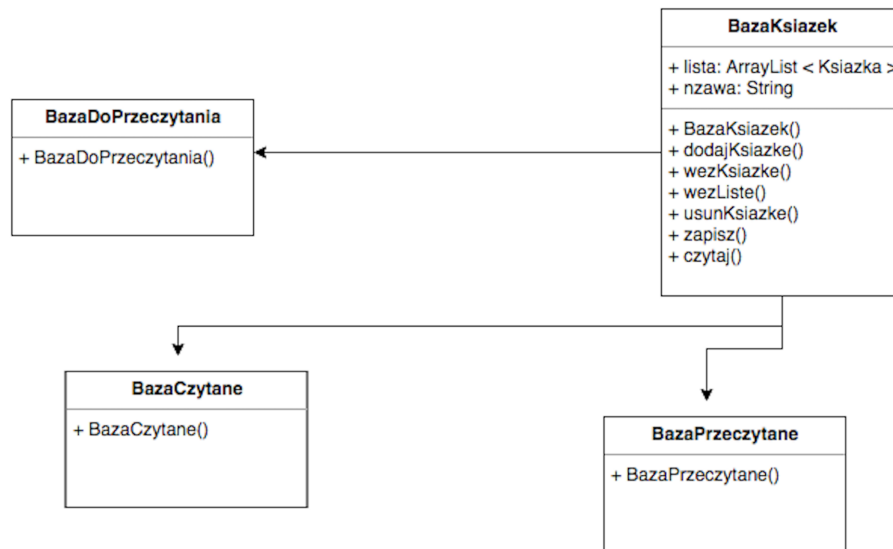
Klasa dziedzicząca po *BazaKsiazek* trzymająca bazę książek do przeczytania.

### 3.1.13 BazaCzytane

Klasa dziedzicząca po *BazaKsiazek* trzymająca bazę książek czytanych.

### 3.1.14 BazaPrzeczytane

Klasa dziedzicząca po *BazaKsiazek* trzymająca bazę książek przeczytanych.



Rysunek 10: Schemat klas dziedziczących po **BazaKsiazek**.

### 3.1.15 MojPrzycisk

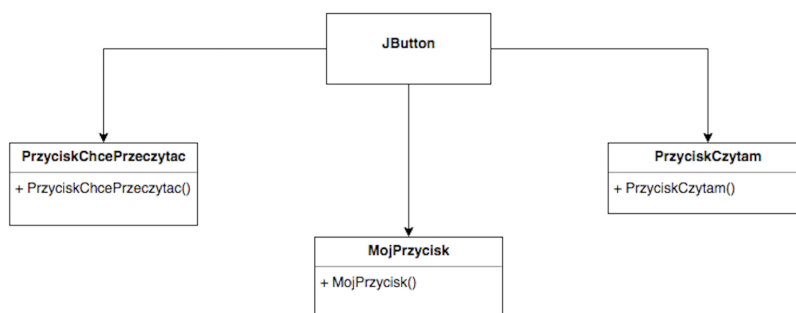
Klasa dziedzicząca po *JButton*. Pomocnicza klasa definiująca przycisk, jego działanie i rozmieszczenie w oknie w zależności od parametrów w konstruktorze.

### 3.1.16 PrzyciskChcePrzeczytac

Klasa dziedzicząca po *JButton*. Tworzy przycisk dodawania książki do bazy książek do przeczytania.

### 3.1.17 PrzyciskCzytam

Klasa dziedzicząca po *JButton*. Tworzy przycisk dodawania książki do bazy książek czytanych.



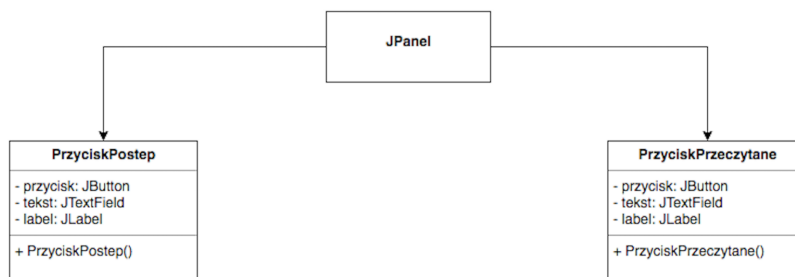
Rysunek 11: Schemat klas dziedziczących po **JButton**.

### 3.1.18 PrzyciskPostep

Klasa dziedzicząca po *JPanel*. Tworzy przycisk który po podaniu liczby stron aktualnie przeczytanych i kliknięciu podaje nam nasz obecny postęp czytania danej książki.

### 3.1.19 PrzyciskPrzeczytane

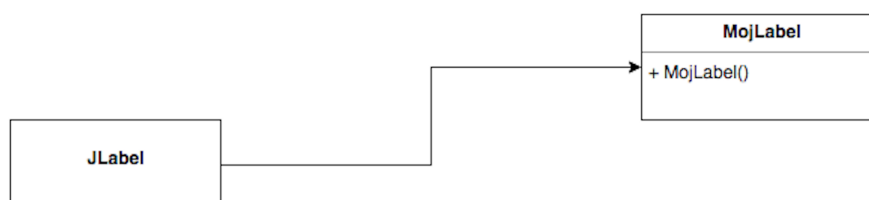
Klasa dziedzicząca po *JButton*. Tworzy przycisk dodawania książki do bazy książek przeczytanych z możliwością ocenienia książki.



Rysunek 12: Schemat klas dziedziczących po *JPanel*.

### 3.1.20 MojLabel

Klasa dziedzicząca po *JLabel* definiująca wszystkie pola tekstowe i obrazkowe w aplikacji.



Rysunek 13: Schemat klas dziedziczących po *JLabel*.

## 3.2 Realizowana funkcjonalność

### 3.2.1 Użytkownik

Każdy użytkownik ma równy dostęp do aplikacji. W aplikacji nie zostały umieszczone żadne ukryte możliwości, więc użytkownik może korzystać z wszystkich funkcji podanych w opisie programu.



### 3.2.2 Programista

Aplikacja jest przystosowana do dalszego rozwoju. Więcej o tym w rozdziale **Dalsza rozbudowa**. Poza rozszerzeniem kodu programista może dowolnie dodawać książki do bazy.

## 4 Dalsza rozbudowa

Aplikacja była pisana w dość krótkim czasie, istnieje zatem wiele możliwości rozwoju. Jedną z ważniejszych rzeczy do rozbudowy jest panel umożliwiający użytkownikowi dodanie własnej książki do bazy w przypadku gdy dana książka jeszcze w niej nie istnieje. Możliwe jest także wprowadzenie różnego rodzaju filtrowania bazy książek (po gatunku, autorze, roku wydania).